Summer 1997

Application of Active Noise Control to Reduce Cabin Noise in Single Engine General Aviation Aircraft

Jeroen Hendrik Dolmans
Embry-Riddle Aeronautical University - Daytona Beach

APPLICATION OF ACTIVE NOISE CONTROL TO REDUCE CABIN NOISE IN SINGLE ENGINE GENERAL AVIATION AIRCRAFT

by
Jeroen Hendrik Dolmans

A Thesis Submitted To The
Aerospace Engineering Department
In Partial Fulfillment Of The Requirements For The Degree Of
Master Of Science In Aerospace Engineering

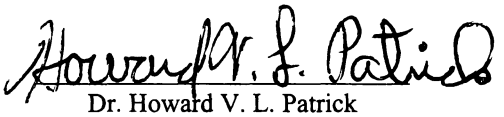
Embry-Riddle Aeronautical University
Daytona Beach, Florida
Summer 1997

APPLICATION OF ACTIVE NOISE CONTROL TO REDUCE CABIN NOISE IN SINGLE ENGINE GENERAL AVIATION AIRCRAFT

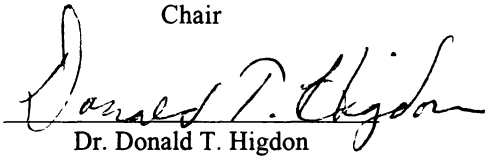
by
Jeroen Hendrik Dolmans

This thesis was prepared under the direction of the candidate's thesis committee chair, Dr. Howard V. L. Patrick, Department of Aerospace Engineering, and has been approved by the members of his thesis committee. It was submitted to the Department of Aerospace Engineering and was accepted in partial fulfillment of the requirements for the degree of Master of Science in Aerospace Engineering.

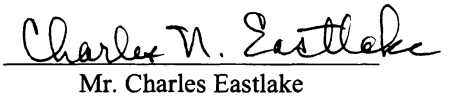
THESIS COMMITTEE:



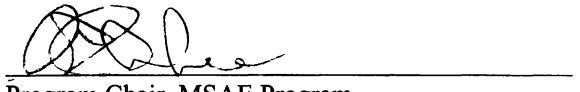
Dr. Howard V. L. Patrick
Chair



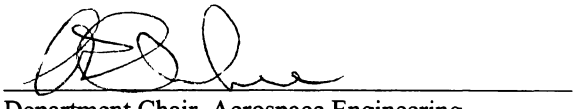
Dr. Donald T. Higdon
Member



Mr. Charles Eastlake
Member



Program Chair, MSAE Program



Department Chair, Aerospace Engineering

7/29/97
Date

ACKNOWLEDGMENTS

This thesis was made possible by the support of Piper Aircraft. I would like to thank Mr. Carlos Latoni, Mr. William Moreu and the rest of the good folk at Piper.

Sincere thanks to my thesis advisor Dr. Howard Patrick for his guidance and advice. I especially value the freedom with wich he allowed me to pursue this work. Thanks to Dr. Donald Higdon of DTH Inc. for his invaluable insight, guidance and assistance with all aspects of this thesis. To Professor Charles Eastlake for taking me flying a few times in his Cherokee to collect data, talk aircraft and fly lazy eights just for the fun of it. Thanks to Dr. Kim for his assistance with all aspects of the graduate student life and for taking me flying to collect Cessna data. To Michael Potash, for making life so much easier with his cheerful help in designing and building the filters and all the other things that run on smoke. Thanks to Dr. Ormsbee for his support and reading the thesis.

Thanks to Dr. Hazen and Monica Frappier for their support and encouragement. To my instructors in Prescott, thanks and keep up the good work. To my friend How Meng Au, thanks for the moral support over the years and also the selfless assistance in preparing this report. Thanks to my friends and fellow graduate students for the fun times and support.

Special thanks to Tom Hall, to Yme and Kim Dolmans and most of all to my parents.

ABSTRACT

Author: Dolmans, Jeroen Hendrik

Title: Application of Active Noise Control to Reduce Cabin Noise in Single Engine General Aviation Aircraft

Institution: Embry-Riddle Aeronautical University, Florida, USA

Degree: Master of Science in Aerospace Engineering

Year: 1997

The application of active noise control to reduce cabin noise in single engine, general aviation aircraft is investigated through the use of the 'filtered x' least mean square algorithm and a simple acoustic feedforward method to generate a reference signal is tested. The system is designed to utilize one reference signal and up to two feedback signals and two audio speakers. The feedforward system consists of a microphone placed in close proximity to the front windshield and isolated from the cabin noise. Cabin noise and reference signals are recorded during flight in a Cessna 172 Skyhawk, a Piper Cherokee 140 and a Piper Malibu Mirage. The recorded data is used in laboratory tests to evaluate the capability of the control system to reduce the cabin noise signal with the recorded reference signal. The reference signal was found to lack coherence with the cabin noise in most aircraft which limited the noise reductions. Alternative feedforward methods are investigated and an alternative reference signal is tested in a laboratory simulation. The results with the recorded data and the modified reference signal are detailed in each case.

TABLE OF CONTENTS

Acknowledgments iv

Abstract v

List of Figures ix

List of Symbols xi

CHAPTER 1 INTRODUCTION 1

1.1 Statement of Problem 1

1.2 Previous Research 2

1.3 Current Research 4

CHAPTER 2 THEORY 6

2.1 Single Engine, General Aviation Aircraft Cabin Noise 6

2.2 Introduction to ANC 9

2.3 Introduction to Sound In Enclosures 12

2.3.1 Rectangular, Hard-Walled Enclosures 12

2.3.2 Aircraft Cabins as Enclosures 17

2.4 Tonal Versus Broadband Control 19

2.5 Algorithm Selection 20

2.6 Introduction to FIR Filters and ANC Algorithms 24

2.6.1 Adaptive Linear Combiners 24

2.6.2 Performance Function 26

2.6.3 Single Input, Single Output, LMS Algorithm 30

2.6.4 Multiple Input, Multiple Output, Filtered X LMS Algorithm 31

2.6.5 Secondary Path Identification 34

2.7 Math Modeling 37

2.7.1 Simulations 37

2.7.2 "A" Weighted Response 43

2.8 Reference Signal Considerations 45

2.9 System Configuration 48

CHAPTER 3 EXPERIMENTAL 50

3.1 Board Control and Algorithm Programming 50

3.1.1 Board Control 50

3.1.2 Algorithm Programming and Circular Buffers 52

3.2 Filters 53

3.2.1 Anti-Aliasing Filters 56

3.2.2 Smoothing Filters 56

3.2.3 Precision Voltage Reference 56

3.3 Microphones 57

3.4 Amplifiers and Speakers 60

CHAPTER 4 TESTING AND ANALYSIS 61

4.1 Electric Simulation 61

4.1.1 Electric Simulation Results and Analysis 62

4.2 Data Collection 66

4.3 Data Analysis 66

4.4 Acoustic Testing 71

4.4.1 SISO Acoustic Testing 71

4.4.2 SISO Acoustic Testing Results and Analysis 74

4.4.3 MIMO Acoustic Testing 76

CHAPTER 5 RESULTS 78

5.1 SISO System Testing Results 78

5.2 SISO High Coherence Testing Results 79

5.3 MIMO Testing Results 83

CHAPTER 6 CONCLUSIONS 87

CHAPTER 7 RECOMMENDATIONS 89

CHAPTER 8 REFERENCES 93

APPENDIX A: Math Simulations 95

APPENDIX B: Equipment Listing 109

APPENDIX C: 'C' Code Listing 110

APPENDIX D: Filter Circuit Diagram 127

APPENDIX E: Additional Literature Resources 129

LIST OF FIGURES

Figure	Page
Figure 2.1.1 Cabin noise spectrum in a Piper Saratoga II HP during cruise	7
Figure 2.3.1 Total acoustic potential energy in an enclosure when excited by a single, pure tone monopole source of variable frequency placed in one corner of the enclosure (solid curve). Dashed curve is the residual total acoustic potential energy in the enclosure when the secondary source is attempting global noise reduction	16
Figure 2.5.1 Simple ANC system block diagram	21
Figure 2.6.1 Single input adaptive linear combiner	25
Figure 2.6.2 Adaptive linear combiner with single input, desired response and error	28
Figure 2.6.3 ANC system block diagram showing secondary path transfer functions and delay	32
Figure 2.6.4 Block diagram of LMS algorithm implementation to perform secondary path identification	36
Figure 2.7.1 Time domain feedback and error data from system identification math model	38
Figure 2.7.2 Simulated cabin noise spectrum before control and residual noise during control	39
Figure 2.7.3 Simulated cabin noise and tonal reference signal	40
Figure 2.7.4 Simulated cabin noise before control and residual noise during control	41
Figure 2.7.5 Simulated cabin noise and two tonal reference signals	42
Figure 2.7.6 Simulated cabin noise before control and residual noise during control	42
Figure 2.7.7 "A" weighted relative response spectrum	43
Figure 2.7.8 Resulting spectrum after applying "A" weighting relative response spectrum to Cessna 172 cabin noise recorded during cruising flight	44
Figure 2.9.1 ANC System block diagram	49
Figure 3.3.1 Comparative test results for the Electret microphone and the B&K microphone	59

Figure 3.3.2 Schematic to illustrate the feedforward system and implementation 59

Figure 4.1.1 System identification impulse response obtained during an electric system simulation 63

Figure 4.1.2 System identification secondary path response transfer function gain 63

Figure 4.1.3 Spectra at feedback sensors 0 and 1 before controlling a filtered ramp function and the residual noise during control 64

Figure 4.1.4 Controller transfer function gain 65

Figure 4.3.1 Cessna Skyhawk 172 cabin noise and reference signal spectra during cruise 67

Figure 4.3.2 Piper Cherokee 140 cabin noise and reference signal spectra during cruise 68

Figure 4.3.3 Piper Cherokee 140 cabin noise and reference signal spectra during climb 69

Figure 4.3.4 Piper Malibu Mirage cabin noise and reference signal spectra during cruise 70

Figure 4.3.5 Piper Malibu Mirage cabin noise and reference signal during climb 71

Figure 4.4.1 Cessna 172 cabin noise spectra before control and residual noise during control 74

Figure 5.1.1 Spectra of Cessna 172 cabin noise before control and residual noise during control 79

Figure 5.2.1 Piper Cherokee cabin noise spectrum before control and residual noise during control with high coherence reference signal 80

Figure 5.2.2 Cessna 172 cabin noise spectrum before control and residual noise during control with high coherence reference signal 81

Figure 5.2.3 SISO secondary path response transfer function gain 82

Figure 5.3.1a. Reference signal and cabin noise spectra at each of two feedback microphones 84

Figure 5.3.1b. Cabin noise spectrum at microphone number 1 before control and residual noise during control 85

Figure 5.3.1c. Cabin noise spectrum at microphone number 2 before control and residual noise during control 85

Figure 7.1 Block diagram to illustrate a multiple tonal reference method through the use of multiple fxLMS controllers 90

LIST OF SYMBOLS

A	Amplitude
B	Amplitude
C'(f)	Secondary path transfer function model
c	Speed of sound
d	Desired value, delay
e	Error signal
f	Frequency
f_{BPF}	Blade passage frequency
f_c	Corner frequency
f_{engine}	Engine speed
f_N	Nyquist frequency
f_n	Frequency of n'th mode
f_s	Frequency at corner of stopband
g	Filtered residual noise signal
k	Sample number, time index
L	Total perimeter length
l	Path length
N	Number of normal modes, filter order
n	Integer multiplier, number of bits
n_x	n'th mode in x direction
n_y	n'th mode in y direction
n_z	n'th mode in z direction

P	Cross correlation vector
P(f)	Primary acoustic path transfer function
p	Sound pressure level
Q(f)	Electric/acoustic transfer function due to feedback signal path
Q_{rms}	Attenuation
q	Voltage resolution
R	Input correlation matrix
r	Filtered reference signal
T	Period
V	Volume of enclosure
$V_{q\ noise\ level}$	RMS quantization noise level
V_{ref}	Reference voltage, voltage range
W	FIR weight array
W^*	Wiener weight vector
w	FIR weight value
X	Sampled reference signal array / FIR input value
x	Reference sampled value
y	FIR output value

Greek and Math Symbols

∇	Gradient
β	Number of propeller blades
ε	Error value

ϕ	Phase angle
$\Gamma(f)$	Electric/acoustic transfer function due to control signal path
λ	Wavelength
$\rho(f)$	Primary sound pressure level
ω	Angular frequency
ξ	Mean-square error

CHAPTER 1
INTRODUCTION

This thesis describes research investigating the application of active-noise-control (ANC) systems to reduce cabin noise in single engine, general aviation (GA) aircraft. This chapter introduces the motivations for the research, provides a brief outline of the background and history of ANC research and details the goals of the research presented in this thesis.

1.1 STATEMENT OF PROBLEM

Single engine GA aircraft cabin noise consists of the air-borne and structural-borne sound generated by the engine, the propeller and the noise due to the aircraft structure interacting with the boundary layer. The noise levels in these aircraft is generally high during both cruise and climb (90 dB -115 dB). The noise is dominated by the low frequency sounds below 300 Hz due to the engine and propeller tones. Cabin noise is fatiguing and makes conversation difficult or unintelligible. The dominant tones, usually due to the propeller, are shown to be the most fatiguing content in the spectrum (Ref. 1) but these tones are the hardest to control by traditional, passive methods. Reducing this low frequency cabin noise by passive methods is not practical in these aircraft but constitutes the frequency range in which ANC is the most effective.

Pilots and passengers in GA aircraft typically rely on passive headsets incorporating an intercom system to provide hearing protection, reduce fatigue and enable conversation. While these headsets provide good noise attenuation, the tones are still present and fatiguing. More recently, active headsets incorporating ANC systems provide an improvement over the passive headsets by eliminating the dominant tones and reducing the broadband component. The use of headsets is cumbersome, restrictive and uncomfortable after extended periods due to 'head squeeze'. ANC aims to reduce cabin noise to provide lower noise levels to allow normal conversation and eliminate the need for headsets. ANC systems eliminate the dominant tones and can provide extensive broadband reductions in some cases.

ANC systems are currently used in some twin turboprop aircraft and twin engine GA aircraft but the cost of these systems is far too high to allow implementing such systems in single engine GA aircraft (Ref. 2). However, digital signal processing (DSP) system costs are continually reducing and it is just a matter of time before such systems become economically feasible for the larger, more complex single engine aircraft and eventually single engine trainers.

Single engine GA aircraft present a unique set of problems due to the aircraft configuration and the restrictive cabin space and weight allowances. The amount of noise reduction due to ANC in such aircraft is also unknown. It is possible to investigate practical considerations as well as the amount and the spatial extent of noise reductions through the use of a notebook computer based ANC system in laboratory and flight tests.

1.2 PREVIOUS RESEARCH

The reduction of low frequency sound and vibration by passive methods is difficult because the long wavelengths result in very thick, or heavy, isolation and damping material. The resulting structure is too massive or heavy in most situations. Active control systems target these frequencies and are the most effective at low frequencies. The basic principle of ANC is to apply a secondary acoustic wave to the unwanted primary acoustic wave (the noise) to cancel the noise. Linear superposition shows that if the secondary wave is of equal amplitude and 180° out of phase with the primary, the primary wave is canceled. The amount of reduction is determined by the accuracy of the control system in generating the secondary wave with the correct phase and amplitude, as well as time-response characteristics.

The first patent for active noise control was applied in 1933 in Germany, and then in 1936 in the USA, by Paul Lueg (Ref. 2). His system consisted of a microphone placed upstream of a controlling speaker. The signals from the microphone were fed through an electric control system to change the phase and amplitude of the sound emitted from the speaker. The acoustic pressure produced by the speaker was to be the inverse of the sound field in the duct, reducing the acoustic pressure in the duct and the duct opening. Due to the limitations of electronic systems at the time, it was not until the 1950's before any significant advancement was made

by Olson (Ref. 3). Olson did an extensive amount of work to explore ANC in rooms, ducts and headsets by the application of feedback control.

Research stalled again due to electronic and transducer limitations until the 1970's and 80's. Advances in control theory and microelectronics enabled progress in the control of low frequencies but commercial applications were still prohibitively expensive in all but a few situations. High power, stable, low-frequency sound and vibrations transducers were still the main limitations together with the cost of digital control systems.

The increasing digital processing power and reducing costs spurred development in the 1980's with approximately 2,200 published technical papers in the 1980's alone. The closely linked fields of ANC and vibration control led to equally extensive work in active-vibration-control (AVC) systems to reduce the vibrations causing the noise.

Modern ANC and AVC systems are characterized by using secondary sources to apply control to the noise, or vibrations, generated by the primary source(s). The control signals driving the secondary sources are generated by a digital controller which implements feedback control through the use of the measured residual noise or vibration. Feedforward systems which obtain a measure of the incoming primary noise or vibration are commonly implemented because such systems generally have improved performance over systems which use only feedback control. These modern systems utilize both feedforward and feedback such that the system is adaptive . Adaptive controllers are capable of automatically tuning the control applied to the feedforward signal to adapt to time varying changes in the system under control.

The first commercially available active control systems were those for reducing low frequency planar sound waves propagating in air ducts. Active control to reduce low frequency noise in industrial systems, air conditioning systems, transformer noise and machine noise has been successfully applied. Systems have also been applied to all sorts of vehicles ranging from trucks to submarines to aircraft, reducing either interior (receiver) noise or exterior (source) noise. Active control systems have also been researched and/or applied to reduce sway in tall

buildings, steady microscopes, improve telescope resolution through the use of adaptive optics and active suspension systems for vehicles (Ref. 3).

Part of the reason ANC and AVC has been difficult to develop on a commercial basis is due to the multidisciplinary nature of the work and the very different system, and sub-system, design requirements for each application. An interesting side note, and warning, are the extensive patent infringement battles (Ref. 3) which have surely raised the cost of commercial systems.

With regard to ANC in aircraft, several systems are currently in production. The Saab 2000 and 340B, and the de Havilland Dash 8 are examples of turboprop aircraft in which ANC systems are implemented to reduce cabin noise. Systems are available from Lord Corporation (Ref. 4) as Supplemental Type Certificate (STC) retrofits in the Beech King Air, Cessna Conquest and Turbo-Commander series of aircraft. Lord Corporation also produces the active vibration control systems implemented in the Cessna Citation X engine mounts as original equipment and received a FAA STC in early 1997 to approve installation of their active engine mounts in the McDonnell Douglas DC-9/MD-80 series. Some of these systems are outlined later in this report but the system details and algorithms are proprietary and are not available. The theory section contains more detailed information and references to previous research which is used in the work reported in this thesis.

1.3 CURRENT RESEARCH

There are several goals in the current research which are outlined in this section. For completion, other areas which are not included in this research are also outlined.

The goal is to record cabin noise and other data during flight in several aircraft to use in laboratory testing to evaluate the amount of noise reduction possible in representative aircraft. The Piper Malibu Mirage, Piper Cherokee 140 and Cessna 172 aircraft are used as representative aircraft in which data is recorded during flight. The data is used during laboratory testing to evaluate the system performance. Only the amount of reduction is investigated and not the spatial extent of the reductions throughout the testing enclosure. The shape and structure of each aircraft cabin determines the acoustic response of the cabin and has

a large effect on the spatial extent of the resulting noise reductions. Since the system is tested in the laboratory and not in these aircraft, the extent of the reductions achieved is not representative of results in the actual aircraft.

To achieve global or widespread noise reductions throughout the volume of the enclosure, is necessary to analyze the acoustic response of the enclosure. An experimental survey of the enclosure response is also necessary to verify the analysis. Such an analysis and experimental survey of the candidate aircraft and the testing enclosure is not performed as it is beyond the scope of this thesis.

The current system is developed to the stage where two speakers provide the controlling sound and two microphones provide feedback data about the residual sound levels. The control system uses both feedforward and feedback data. A single, acoustic feedforward transducer is used to generate a single reference signal.

This research is intended to provide the groundwork for future work to develop the system to the stage of flight testing. To fulfill this role, the present report contains a theoretical introduction to ANC algorithms, reference signal considerations and math modeling. The results achieved with the current system are presented and analyzed. The recommendations at the end of this report detail system development and areas of concentration for future research.

The next chapter presents some of the theory necessary to developing the system. The first section presents an analysis of the cabin noise in the type of aircraft considered here which is necessary to determine the best control strategy.

CHAPTER 2
THEORY

This chapter presents an overview of the cabin noise in the type of aircraft under consideration, ANC methods and the types of ANC. The selected method is outlined and an overview of sound in enclosures is presented as it relates to the system. ANC algorithms and digital filtering theory are introduced followed by the derivation of the implemented algorithm. Mathematical modeling is presented as a tool to understand the behavior of the algorithm and the expected system performance. Reference signal considerations and the final system configuration are the final sections of this chapter.

2.1 SINGLE ENGINE, GENERAL AVIATION AIRCRAFT CABIN NOISE

This section provides a brief introduction to the noise sources in general aviation aircraft cabins. The sound and vibration propagation paths which result in the cabin noise are also introduced. Figure 2.1.1 (Ref. 5) shows the sound-pressure-level (SPL) spectrum during cruise as measured in a Piper Saratoga cabin and is a representative example of the typical cabin noise spectra for the type of aircraft considered in this research. There are numerous tones above a large broadband component. The sources of the broadband and tonal components are discussed in the following paragraphs.

The cabin noise consists of the sum of the sounds transmitted into the cabin by the engine, propeller and airframe. The airframe noise is primarily low amplitude, broadband random noise due to the boundary layer exciting the fuselage skin. Transient airframe noise such as control surface actuation is neglected. The airframe noise is generally negligibly small compared to the other sound sources.

The propeller noise consists of both tonal and broadband components. The tonal components are dominant and occur at the blade passage frequency (BPF) and harmonics of BPF, and propeller imbalance if present. These tones are due to blade loading and thickness noise. The broadband

noise is caused by the vortex shedding from the trailing edge of the propeller blades (Ref. 6). The loudest tone corresponds to the BPF which is easily determined as follows:

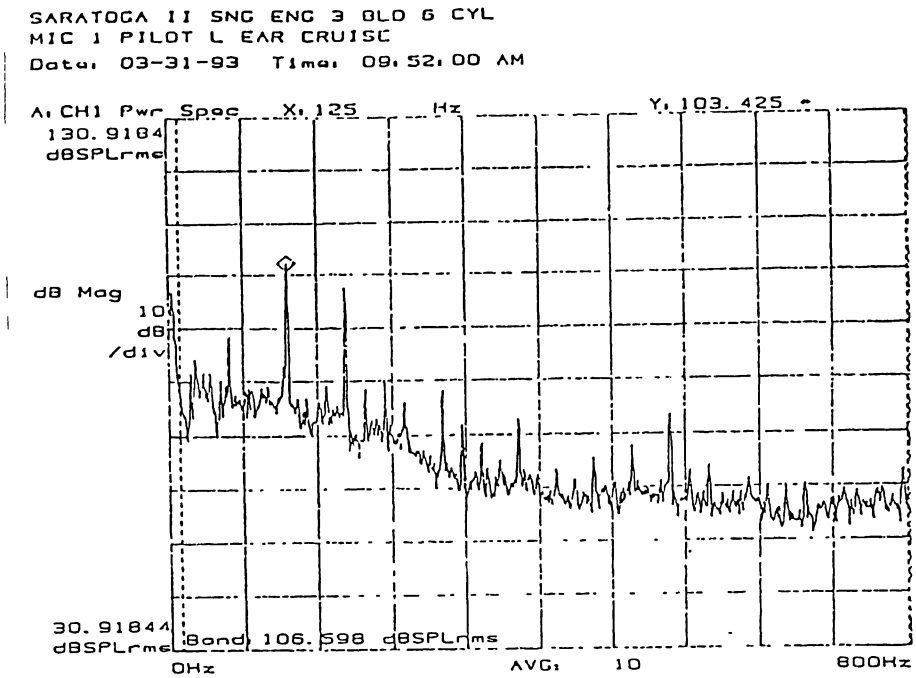


Figure 2.1.1 Cabin noise spectrum in a Piper Saratoga II HP during cruise (Ref. 5)

$$f_{BPF} = \frac{f_{engine}}{60} \cdot \beta \tag{2.1}$$

where f_{BPF} is BPF in Hz, f_{engine} is the engine speed in cycles per minute (rpm) and β is the number of blades. A typical general aviation aircraft piston engine turning at 2,400 rpm during cruise has an 80 Hz BPF with a 2 bladed propeller. The harmonics of BPF are integer multiples of BPF with the first harmonic at twice BPF (160 Hz), the second harmonic at three times BPF (240 Hz) and so on.

The Saratoga cabin noise spectrum is for an aircraft fitted with a 3 bladed propeller and a six cylinder engine operating at 2500 rpm. This results in the fundamental BPF tone at 125 Hz BPF. There is another significant tone at approximately 190 Hz which is due to exhaust noise (Ref. 5).

The engine noise consists of both tonal and broadband noise. The tonal components have frequencies proportional to the speed of the rotating and reciprocating components such as pistons, gears, valves, pumps and due to the firing frequency. During climb engine noise is dominated by the pulsating (monopole) source at the exhaust.

The total acoustic spectrum in the cabin of a single engine, propeller driven aircraft is composed of the spectrum excited by both the acoustic paths and the structural paths. The acoustic path consists of the propeller pressure fluctuations impinging on the both the front and sides windshields, and the fuselage skin, causing them to vibrate. These vibrations cause pressure fluctuations in the air inside the cabin, producing the cabin noise corresponding to the acoustic path. The structure borne path consists of vibrations due to the propeller and engine passing through the engine mounts, exciting vibrations in the fuselage structure and firewall which excites the corresponding cabin noise. Additional structural borne sound can be due to the landing gear and wing struts attachments but these are usually negligible.

Since the loudest components in the sound spectrum occur at BPF, and associated harmonics, the acoustic path is expected to be the dominant path in general aviation, single engine aircraft with a tractor propeller configuration. The contribution by each path to the total sound spectrum is difficult to determine but the structural borne path is shown to be the lesser of the two (Ref. 7) in at least one general aviation aircraft (Cessna 172 Skyhawk) by evaluating the coherence between the cabin noise, the windshield vibrations, and engine mount vibrations. The research indicates the dominant sound transmission path is the acoustic path through the windshield vibrations induced by the propeller. The contribution by the vibrating fuselage skin are not evaluated in the research.

A knowledge of the dominant contributors to the cabin noise and the sound transmission paths is necessary because these are driving factors in the design of an effective noise control system. The following section introduces different ANC strategies and the current system configuration.

2.2 INTRODUCTION TO ANC

This section contains a brief introduction to the different methods which are used, or tested, to actively reduce cabin noise. Where possible, the overall reduction achieved in practical implementations of each method are outlined. These results allow the reader to develop a realistic feeling for the amount of reduction possible, the performance goals of this research and the reasons for the selection of the methods used in this research. This introduction provides a very brief overview of the methods used and is not a detailed discussion of the technical aspects involved. The references and additional literature resources listed in the appendix are provided for those readers who want to read about ANC in more detail.

The final goal of ANC is to reduce the unwanted sound (noise) and most methods use a number of feedback microphones as error sensors for the control system. The mechanism by which the sound field is excited in the cabin is usually the control system design driver. Control can be applied acoustically, through the use of audio speakers, or by techniques which try to apply control at, or closer to, the source of excitation through the use of active vibration control (also referred to as active structural acoustic control, ASAC).

AVC is applied in situations where the sound transmission path from the source to the cabin is primarily through structural vibrations. ASAC aims to control those vibration modes in the structure which are the dominant contributors to the acoustic excitation in the cabin. As vibrations travel from the power plant to the cabin, numerous higher order modes are excited to levels where they become significant contributors to the cabin noise. By preventing the vibrations close to the source, ASAC prevents the excitation of higher order modes in addition to those being controlled. This technique results in good global cabin noise reduction because it prevents the transmission of vibrations from the power plant to the cabin structure. Feedback microphones are still used in the cabin and in some cases accelerometers are used in conjunction with the feedback microphones. The three most promising forms of ASAC in aircraft take the form of active engine mounts, active vibration absorbers and piezoceramic transducers.

Active engine mounts are actively controlled transducers which are used to mount the engines to the structure in place of, or in conjunction with, the usual engine mounts. The mounts must be

rigid and fail-safe to provide the require levels of engine restraint. The active mounts must be able to provide the required structural compliance and generate the canceling forces necessary to control the vibrations. Active engine mounts are most useful in aircraft where the structural borne engine noise is very large compared to the direct, acoustic path. Examples of aircraft which benefit the most from active engine mounts are those with wing mounted, or rear pylon mounted, turbofan engines. Typical examples of such aircraft are business jets and aircraft configured such as the DC-8/MD-80. By reducing the vibrations at the bottle necks created by the engine mounts, the engine vibration is effectively isolated from the fuselage and noise reductions are global to fuselage.

Lord Corp. tested such a system on a twin engine business jet by replacing the two forward engine mounts with active mounts. These mounts generated control forces in two directions and test flight showed 16 dB reduction in cabin noise and 20 dB reduction in vibration levels (Ref. 4). The Cessna Citation X uses similar mounts from Lord Corp. as standard equipment (Ref. 4).

Tuned vibration absorbers (spring-mass-damper systems) attached alongside some of the fuselage frames and stringers at strategic locations behind the interior trim panels are effective for reducing cabin noise in certain aircraft types. This method is most suitable where the noise transmission is primarily through vibrations which are induced in fuselage at fairly localized regions. Aircraft with wing mounted engines cause the propellers to excite strong, localized fuselage vibrations due to the directivity pattern of the acoustic pressure fluctuations (flanking) caused by the propellers. In such twin engine aircraft the flanking is the primary noise propagation path into the fuselage and is quite controllable. In previous installations where the vibrations absorbers are passive (i.e. not adaptive), the spring-mass-damper system is optimized for cruise conditions. This means that noise reduction is only realized at the propeller speeds at which the vibration damper is optimized. By using active (adaptive) vibration absorbers, noise reduction are possible at different propeller speeds. Tuned vibration absorbers reduce noise and vibration due to the fundamental BPF tone and some higher-order harmonics. Engine noise cannot be controlled but this is not the dominant noise in turboprop aircraft. The deHavilland Dash 8 incorporates such a system to reduce cabin noise induced by the twin, wing mounted turboprops (Ref. 4).

Lord Corp. has tested such a system by installing 8 active vibration absorbers on a twin turboprop fuselage. Speakers were used to simulate the flanking and up to 14 dB noise reduction and 18 dB reduction in vibration levels resulted in the cabin (Ref. 4).

Piezoceramic actuators on the fuselage skin are also used (Ref. 8, 9, 10). The principle is to apply forces directly to the vibrating skin to reduce the excitation of the cabin noise. Extra care must be taken with ASAC if the actuator size is very small compared to the wavelength of the frequency to be controlled. Such an actuator can excite higher frequency modes due to control energy spillover. The result may be that the noise level is reduced but many higher frequency modes are excited which has structural fatigue implications. In an effort to reduce the control energy spillover, groups of piezoceramic patches applied to the fuselage skin are tested with some success in at least one experiment (Ref. 8) on a DC-9 fuselage.

The acoustic spectrum in the type of aircraft considered here is dominated by the acoustic transmission path via the windshields and not the structural borne path through the engine mounts and fuselage structure. Active engine mounts have less potential in this application and applying active vibration absorbers to the windshield supporting structure is not feasible for this research and will only have limited effect. Acoustic ANC with the control signal supplied through the use of audio speakers is selected for implementation for the reasons outlined above. A feedforward method consisting of a single microphone placed in close proximity to the front windshield, and acoustically isolated from the cabin, is used to generate the reference signal used by the control algorithm.

Systems using speakers have been successfully implemented in the Saab 9000 by Ultra Electronics and in the Beechcraft King Air, Cessna Conquest and the Turbo-Commander by Lord Corp. (Ref. 4). Lord Corp. (Ref. 4) has achieved reductions of up to 70% (10 dBC) in the King Air 200 with approximately 20 dB reduction of the BPF component. This company has also demonstrated reductions in a helicopter of up to 75% (12 dBA) where speakers are used to simulate the presence of the main rotor (Ref. 4).

Most of these systems use a number of feedback microphones distributed inside the cabin and a feedforward method is used to pass propeller speed information to the control algorithm. Feedforward systems which have been tested are a propeller shaft pickup, strain gages and ignition pulses. The Lord Corp. system for the King Air utilizes an accelerometer in the nose cone to measure the vibration levels induced by the propellers (Ref. 4). The acoustic feedforward system mentioned previously is tested in this research because it has the attraction of being a very simple, low cost system.

To effectively apply acoustic ANC, an understanding of sound in enclosures is necessary. The next section provides the reader with the necessary background information and shows the potential performance of acoustic ANC in enclosures.

2.3 INTRODUCTION TO SOUND AND ANC IN ENCLOSURES

The amount of reduction and the extent of the reduced sound field is in large part dependent on the acoustic response of the enclosure in which the system is implemented. It is not the intent of this research to develop and install the control system in a particular aircraft so a detailed analysis and mapping of the cabin response is not performed. It is instructive to briefly examine the acoustic response of enclosures to obtain a feeling for the effects this has on system performance and to show the importance of a detailed analysis before attempting to install a system in an aircraft. The simple case of a hard-walled, rectangular enclosure is discussed first.

2.3.1 Rectangular, Hard-Walled Enclosures

Consider a simple rectangular enclosure containing a single, monopole source (speaker) radiating sound. The sound will radiate spherically from the source, with the wave front traveling away from the source in all directions. The normal at any point on the wave front indicates the direction of motion taken by that point on the wave front. Visualize the normal to trace out a ray which corresponds to the path taken as the wave front moves outward. Initially there are infinitely many rays traveling outward from the source. As the rays reach the boundaries of the enclosure they are reflected back into the room with reflection angles equal to the incidence angles. The waves are reflected such that some paths are repeated over time by the same rays. An example of this is the

path due to sound reflecting back and forth between two parallel walls. These paths will be repeated in a periodic manner dependent on the length of the path and speed of sound.

In some cases the frequency of the sound is such that after the completion of one trip around the repeating circuit, the radiated and the reflected sound are in phase and will reinforce each other. With each circuit, more energy is pumped into the path. The result is a resonating, standing wave with large amplitude. The standing waves are referred to as the (acoustic) normal modes of vibration of the enclosure. The reflected and radiated sound are in phase if the path length, l, is an integer multiple, n, of the wavelength, λ:

$$n\pi = l \tag{2.3.1}$$

Expressed in terms of frequency through the use of the relationship (where c is the speed of sound):

$$f = \frac{c}{\lambda} \qquad \text{gives} \tag{2.3.2}$$

$$f_n = \frac{nc}{l} \tag{2.3.3}$$

As an example consider the reflections between two parallel walls with the sound traveling in the x direction. The path length is completed twice in one circuit so the resonance frequency is:

$$f_n = \frac{n_x c}{2l_x} \tag{2.3.4}$$

When the sound source radiates a range of different frequencies, many different normal modes are excited in the enclosure. It can be shown through the use of wave theory that the frequencies corresponding to the normal modes of a rectangular, hard walled enclosure must satisfy the following relationship (Ref. 11):

$$f_n = \frac{c}{2}\sqrt{\left(\frac{n_x}{l_x}\right)^2 + \left(\frac{n_y}{l_y}\right)^2 + \left(\frac{n_z}{l_z}\right)^2} \tag{2.3.5}$$

where: $n_x, n_y, n_z = 0, 1, 2, 3, \ldots\ldots, \infty$

l_x, l_y, l_z = dimensions of rectangular enclosure with the walls aligned parallel to the axes

From the above equation notice that more than one mode can be excited by each frequency due to the dimensions of the rectangular enclosure therefore many modes are excited by a source radiating sound containing several frequencies. The resulting normal modes cause standing waves in the enclosure and the acoustic pressure will vary throughout the enclosure as a result of the room response. An approximation of the number of normal modes, N, below the highest frequency of excitation which exist in the enclosure is closely estimated by (Ref. 11):

$$N = \frac{4\pi f^3 V}{3c^2} + \frac{\pi f^2 A}{4c^2} + \frac{fL}{8c} \tag{2.3.6}$$

where: $A = 2(l_x l_y + l_y l_z + l_x l_z)$ = total surface area

$L = 4(l_x + l_y + l_z)$ = sum of all edge lengths

$V = l_x l_y l_z$ = volume of enclosure

At higher frequencies the number of modes increases with the cube of the frequency because the cubic term dominates the other two terms. A general expression to describe the sound pressure distribution in each of the normal modes in an unforced sound field (without a source exciting the sound field) is written as follows for low frequencies (Ref. 11):

$$p(x,y,z) = \sum_{n_x=0}^{\infty}\sum_{n_y=0}^{\infty}\sum_{n_z=0}^{\infty} A_n \cos\left(\frac{n_x \pi x}{l_x}\right)\cos\left(\frac{n_y \pi y}{l_y}\right)\cos\left(\frac{n_z \pi z}{l_z}\right) \tag{2.3.6}$$

Where A_n is a constant associated with each mode. The above equations serve to show that if a source containing many frequencies is radiating sound in an enclosure, the acoustic response of the enclosure modifies the sound field at each point in the enclosure. In the case of ANC, the position of the both the microphones and the speakers effects both the amount and the extent of the sound reduction. If the primary sound source contains a strong tonal component, such as BPF, which coincides with one of the natural frequencies of the room, a large buildup of acoustic energy results at this frequency. This acoustic mode dominates and causes large variations in sound level throughout the enclosure.

The overall acoustic potential energy in an enclosure is proportional to the integral of the mean-squared sound pressure level (SPL) throughout the enclosure. The reduction in the overall acoustic potential energy is one way to measure the effectiveness of a global noise reduction system. A global noise reduction system is one which attempts to reduce the noise throughout the entire volume of the enclosure. An experimental approximation of the total acoustic potential energy is obtained by summing the mean-square pressures obtained from a finite number of monitoring microphones in the enclosure. The microphones are located such that the dominant acoustic modes are detected.

An example of ANC in a rectangular, hard walled enclosure is now outlined to illustrate some of the limitations of acoustic ANC. This example is largely taken from Reference 12. Consider the (1.9m x 1.1m x 1.0m) enclosure shown in Figure 2.3.1 with a single, monopole source in one corner emitting a pure tone. The frequency of the tone is swept from 20 Hz to 300 Hz as indicated by the solid line in the figure. The natural frequencies of the enclosure cause the peaks in the total acoustic potential energy over the 300 Hz range shown on the graph. The mode shape nodes are shown on the walls of the enclosure in the figure by the dotted lines. The natural frequencies tend to clump together around 175 Hz due to the geometry of the enclosure as it relates to the wavelength of the tone.

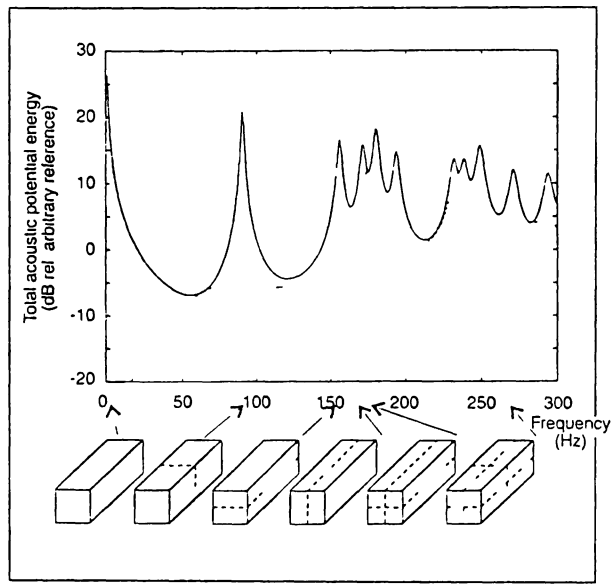


Figure 2.3.1 Total acoustic potential energy in an enclosure when excited by a single, pure tone monopole source of variable frequency placed in one corner of the enclosure (solid curve). Dashed curve is the residual total acoustic potential energy in the enclosure when the secondary source is attempting global noise reduction (Ref. 12)

A secondary monopole source is now introduced in the opposite corner of the enclosure. This secondary source is driven at the same frequency as the primary source. The phase and amplitude of the secondary source are adjusted through the use of a control system to minimize the total acoustic potential energy. The dotted line on the graph shows the residual energy. At low frequencies, below 30 Hz, the secondary source is able to drive an acoustic mode equal and opposite to the primary source, canceling the tone. As the frequency rises, more modes are excited in the enclosure and the secondary source is able to control only a few of the total number of modes. At the first natural frequency (90 Hz) there is one dominant mode which the secondary source can control, substantially reducing the noise level.

Between the low natural frequency modes and at higher frequencies (above 200 Hz) the secondary source is unable to control any one of the modes without exciting some of the other modes, increasing the total acoustic potential energy in the enclosure. For this reason, the amplitude of the secondary source is low and very little noise reduction results at these frequencies.

Increasing the number of secondary sources allows more of the modes to be controlled. However, as shown previously, at higher frequencies the number of contributing modes increases by the cube of the excitation frequency. For this reason an upper frequency of 300 to 400 Hz is the limit at which global noise control is possible.

2.3.2 Aircraft Cabins as Enclosures

A rectangular, hard walled enclosure is not representative of an aircraft cabin. General aviation aircraft cabins are generally not rectangular due to aerodynamic considerations. Aircraft such as the Piper Malibu Mirage and Saratoga are the clearest examples with fuselages resembling tapered cylinders with neither the front windshield, firewall or aft bulkheads perpendicular to the longitudinal axis of the cabin. Both these aircraft are also equipped with extensive damping and insulating treatments. None of the aircraft considered can be described to have hard walled boundaries since the fuselage skin, windows and bulkhead are all know to vibrate and transmit sound into the enclosure. The cabin shape, structure, damping qualities and loading are all contributors to the acoustic response of the cabin resulting in a diffuse field, lightly damped enclosure which is not easily analyzed. Reference 13 is an excellent source for further study into the very important effects of the enclosure response.

An experimental survey of the cabin of each aircraft type is necessary after all acoustic modeling is complete to ensure optimal, practical locations are used for speaker and microphone placement. Such an experimental survey is beyond the scope of this research and is not entirely necessary given the goals of this work.

Several other factors conspire to make global cabin noise control difficult in this application. Unlike the simple example above, the cabin noise is not excited by a simple monopole source. The propeller and engine are distributed sources and enter the cabin through different sound transmission paths discussed earlier. The sound is excited in the cabin through many distributed sources of different intensities and spectral content. It is desirable to use a minimal number of secondary sources and microphones to reduce the sound levels in the cabin which further limits the potential for global control. There is a volume around each microphone where the noise levels are

reduced locally if global control is not achieved. The goal of the system is to reduce the noise levels at each of the microphones and obtain local control in as widespread a volume as possible. By careful selection of the number and placement of the microphones and speakers, it is possible to achieve fairly widespread control around the head locations occupied by each of the passengers.

Space, weight and power draw requirements are all mayor concerns in aircraft of this type. The acoustic response and resulting component placement plays a significant role in each of these system requirements. The simple analysis above shows several important points. Each of the error microphones must be located such that they are influenced by the same acoustic modes as the listeners and the speakers are located to efficiently drive the acoustic modes detected by the listeners. This allows the speakers to effectively reduce the noise with minimal power requirements and the microphones to ensure the largest possible sphere of influence around the heads of the passengers. Minimal power requirements lead to smaller speakers, speaker weights and enclosures, and minimal power draw by the system.

The diaphragm type speakers commonly used are significantly more efficient when installed in a speaker enclosure, particularly at the low frequencies needed. The limited space available for the speaker enclosures is problematic because it has the effect of raising the power requirements and possibly the size of the speaker needed to reproduce the lower frequencies.

Other concerns not investigated are the variation between aircraft of the same model due to manufacturing tolerances and aircraft aging. This can change the acoustic response of the cabin and the noise spectrum in the cabin as excited by the propeller and engine. Pressurized aircraft such as the Malibu Mirage are expected to show minimal variation. These issues require investigation to determine the necessary system flexibility and robustness before the system design is finalized.

Due to the scope and focus of this research, the acoustic response of aircraft cabins are not analyzed or surveyed experimentally. The variability between aircraft of the same model and aircraft aging effects are not investigated. Altitude effects on the cabin acoustic response are not investigated. These are all areas requiring investigation before the system design is finalized and

any flight tests are attempted. Readily available speakers and speaker enclosures are used since the system is not tested in flight or even in a representative enclosure. System tests are only performed in a readily available enclosure in the laboratory as discussed in Chapter 4.

2.4 TONAL VERSUS BROADBAND CONTROL

Tonal control consists of reducing just the dominant tones. Any reduction of the tonal components below the broadband level has a negligibly small effect on the overall noise reduction unless the broadband component is also reduced. For tonal control, the broadband level can be thought of as a datum level to which the ANC system must reduce the tonal components.

Broadband control consists of reducing the entire spectrum, including all dominant tones and broadband components. The broadband component consists of both random and periodic sound. The random components are more difficult to control than the periodic components and require more powerful algorithms and faster processors than is needed for tonal control. Broadband noise reduction accomplished on an experimental scale to reduce road rumble noise in cars is presented in several papers such as Ref. 14 and rocket payload compartment broadband vibration reduction is also tested (Ref. 15).

The residual noise is evaluated in two ways. The overall sound pressure level (OASPL) reduction gives a quantitative measure in decibels of the system performance. The spectral content and resulting sound quality effect the perceived noise reduction. The OASPL reduction can be deceiving when evaluating system performance. Consider a spectrum composed of bandwidth limited (0 Hz to 700 Hz) white noise. If the white noise is reduced by 6 dB throughout the spectrum, the OASPL reduction is only 6 dB. If there is a tone which dominates by 20 dB over the white noise amplitude and this tone is reduced to the broadband level, the OASPL reduction is less than 20 dB. The resulting reduction depends on the OASPL of the white noise component. If the white noise OASPL is 100 dB, the resulting noise OASPL reduction after canceling the tone is approximately 2 dB. The broadband component in cabin noise is not constant with frequency as it is in white noise but has larger amplitudes at lower frequencies. Broadband reductions at these lower frequencies, together with tonal control, provides the largest perceived and OASPL reductions.

The algorithm and feedforward method selected for this research can result in broadband reduction at lower frequencies although this is not the primary objective. This research focuses on tonal noise control because this is where the noise reductions are most noticeable and it is the dominant tones which are the most fatiguing to people. The feedforward method generates a broadband reference signal which is necessary for broadband control and the algorithm is capable of broadband control. Any broadband noise reduction certainly desirable and of interest from a systems point of view. The next section provides an overview of the algorithms considered and explains why the selected algorithm is determined to be the most suitable for this application.

2.5 ALGORITHM SELECTION

This section introduces the performance requirements for the algorithm, a selection of candidate algorithms and the reasons for selecting the algorithm implemented in this work. This section is not a detailed discussion and only provides an overview. The next section presents some of the theory behind digital filtering and contains a derivation of the algorithm selected for use.

The schematic in Figure 2.5.1 shows the primary noise transfer function associated with the plant, P(f), as detected by the feedback microphone. The transfer function associated with the controller, H(f), between the reference signal and the speaker is updated by the adaptive controller, A(f). The adaptive controller utilizes the reference and feedback signals to optimize the transfer function H(f) to reach and maintain a minimal error state. The primary noise path and the acoustic response are both too complex to allow a closed form determination of the optimum controller transfer function. The controller transfer function must also be able to change to compensate for different cabin loading conditions (number and location of passengers) and flight conditions. For these reasons an adaptive controller is needed which modifies the controller transfer function to minimize the noise.

Although the block diagram in Figure 2.5.1 shows frequency domain transfer functions, the algorithm is not necessarily applied in the frequency domain. The optimization of the control transfer function is based on either a frequency domain or a time domain application of the reference and feedback data.

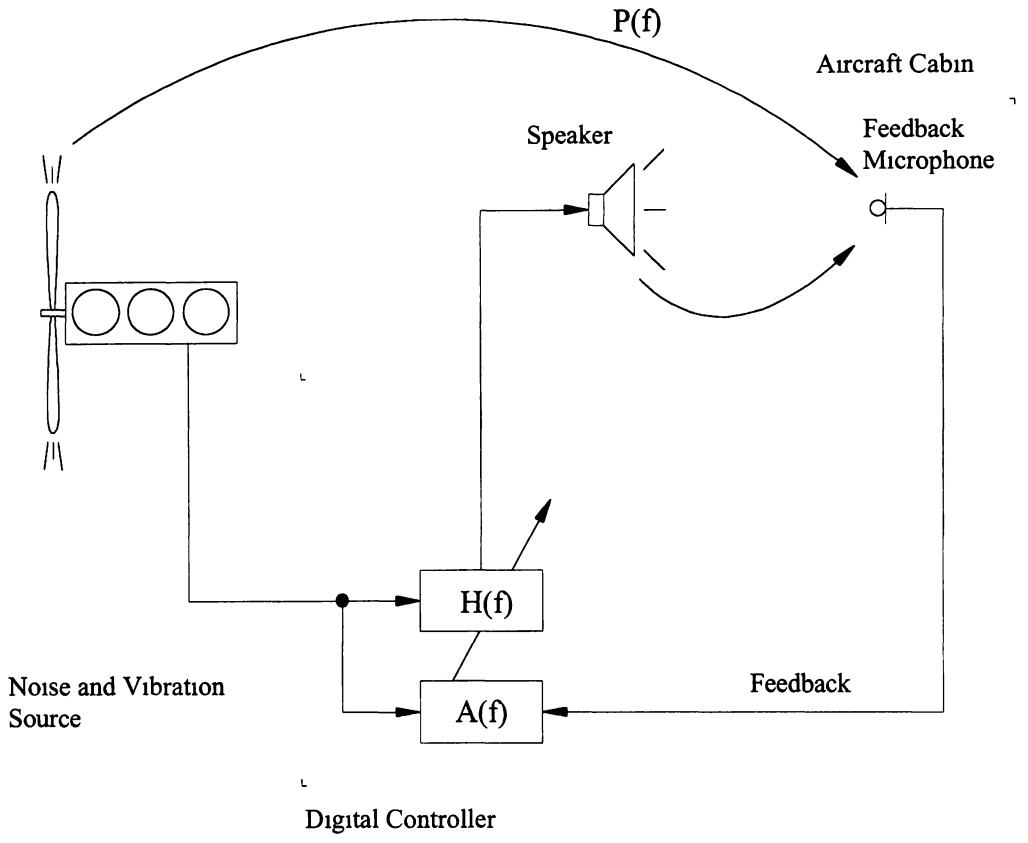


Figure 2 5 1 Simple ANC system block diagram

Both the reference, control and feedback data are in the time domain A frequency based algorithm must convert between the two domains before and after analysis takes place A time domain based algorithm does not require the additional calculations necessary to convert data to the frequency domain and can proceed directly with the optimization This reduced computational load makes time domain algorithms more attractive and these are the first algorithms applied to ANC Time domain algorithms are usually applied in digital control through the use of adaptive linear combiners (also called finite impulse response, FIR, filters) or adaptive recursive combiners (also called infinite impulse response, IIR, filters) FIR methods are the easier of the two to analyze and these were the first which were implemented successfully A basic introduction to adaptive linear combiner filter structures implemented as FIR filters is included in the next section

The feedback data is considered error data since the feedback sensor detects the residual noise during control. It is easily shown that the mean-square error (MSE) is a quadratic performance function of the filter weights (derived in the next section) with only one minimum. The ANC algorithm is reduced to one which seeks the minimum on the performance surface and many adaptive processes do so by gradient search methods.

An efficient algorithm is needed because it must perform all weight updates in real-time. A minimal number of multiplication, division, addition and subtraction operations are desirable. Algorithms avoid the use of matrix inversion, logarithmic functions and other mathematical functions because these operations are computationally intensive and take a large amount of time. The convergence speed and reductions are traded against the amount of calculations needed by different algorithm. More powerful algorithms are generally more computationally intensive.

Due to the constraints of this research, a simple algorithm which yields good tonal noise control with a minimal amount of calculation is the one most suitable. Because tonal noise reduction is the primary focus of this research, a simple algorithm provides the desired performance and satisfies the desire for a minimal amount of calculations performed at each iteration.

A common gradient search method familiar from calculus is Newton's method which uses estimates of the gradient to successively approach the minimum on the performance surface. An alternative to Newton's method is the method of steepest descent in which the weights are adjusted in the direction of the MSE gradient. Neither of these methods are applicable to this type of ANC. The main reasons for this are stability problems which occur near the optimal solution, the problem of accurately determining the gradient and the amount of calculations associated with the gradient determination. However, gradient search methods are attractive because of the simple methods used.

The least-mean-square (LMS) algorithm (Ref. 16) is a well known gradient search method to descend on the performance surface to obtain the minimum mean square error which avoids the problems associated with the method of steepest descent and Newton's method. The LMS algorithm uses the square of the error data as the estimate of the slope of the performance surface

at each iteration. By using this estimate for the slope, the amount of calculation is greatly reduced and stability is ensured by appropriate choice of gain values.

A special form of the LMS algorithm, the 'filtered x' LMS algorithm (fxLMS), is a modification of the LMS algorithm. The algorithm uses the multiple input, multiple output (MIMO) LMS algorithm with the reference signal filtered by a model of the transfer function associated with the secondary path response. This secondary path response model is necessary to account for the frequency response and transport delays between the speakers and microphones.

Assuring a stable system is relatively straight forward and the fxLMS algorithm is applied successfully to control noise in ducts, transformers, automobiles, and aircraft and is a widely used algorithm. Systems implemented in automobiles to control engine booming noise shows convergence times of one tenth of a second (Ref. 12). This algorithm has a large volume of literature available and most of the convergence characteristics are understood. The algorithm is most useful for tonal noise control although some of the lower frequency, broadband noise is also attenuated in some cases.

The algorithm is applied experimentally to broadband noise reduction (Ref. 15) but other algorithms become more efficient and yield better performance due to the very large amount of arithmetic calculations necessitated by the use of the fxLMS algorithm. More powerful algorithms exist which give larger reductions and faster response times than the fxLMS algorithm. The most likely candidates are briefly discussed below. Since the fxLMS algorithm is one of the first algorithms successfully applied to ANC, other algorithms are usually compared to the fxLMS algorithm. The same approach is used here.

Goddard's algorithm is similar to the LMS algorithm except that the convergence parameter (or gain) is also adaptive (Ref. 15). This is a powerful algorithm with faster convergence and larger reductions and can be applied to broadband type noise control. The modified Goddard's algorithm (Ref. 15) uses some simplifying estimates and makes use of the Fast-Fourier-Transform to reduce the number of calculations but still requires more calculations than the fxLMS algorithm.

The 'filtered x' least squares lattice (LSL) algorithm (Ref. 14) yields both faster convergence and lower residual noise than the fxLMS algorithm but requires a quadratic increase in the number of calculations for a given number of weights. The LSL algorithm is applied to control road noise in automobile cabins which is a broadband type noise.

Neural networks show improved performance over the fxLMS algorithm in simulations involving non-linear feedforward control systems but also require significantly larger amounts of calculations (Ref. 14). Non-linear effects are neglected in this research.

The fxLMS algorithm is the best documented, requires the least amount of calculations, and therefore has the lowest hardware requirements of all the algorithms studied. The algorithm is useful for both tonal noise control as well as limited broadband control and provides an excellent starting point for practical investigations into ANC. Because of these advantages the fxLMS algorithm is selected for the research presented.

An introduction to the theory of FIR filters and a derivation of the fxLMS algorithm is presented in the next section.

2.6 INTRODUCTION TO FIR FILTERS AND ANC ALGORITHMS

This section contains an introduction to FIR filters and shows the application of FIR filters to a simple ANC example. The performance function is derived and briefly discussed. The LMS and the fxLMS algorithms are derived and some practical issues are presented.

2.6.1 Adaptive Linear Combiners

The most commonly used non-recursive, adaptive linear combiner is the finite impulse response (FIR) filter. It is a simple, time-varying, linear digital filter whose output is a function of a finite number of past and present input samples only and is not a function of past filter output samples (i.e. non-recursive). The filter is linear because for a given weight vector, the output is a linear combination of past samples of the input vector. When the weights are in the process of adapting, the FIR output is no longer linearly related to the input because the weights are also a function of

the input. For the algorithms used in ANC, the weights are assumed to adjust sufficiently slowly, compared to the input, that the FIR is considered linear.

Reference 16 is used extensively and the notation used here is taken from this reference. Figures 2.6.1 shows a single input adaptive FIR filter implemented in the form of a transversal filter with unit sample delay elements. The input vector consists of the sampled data from the microphones. If k is taken as the time subscript of the sampled data and there are L+1 weights then the input column is as follows:

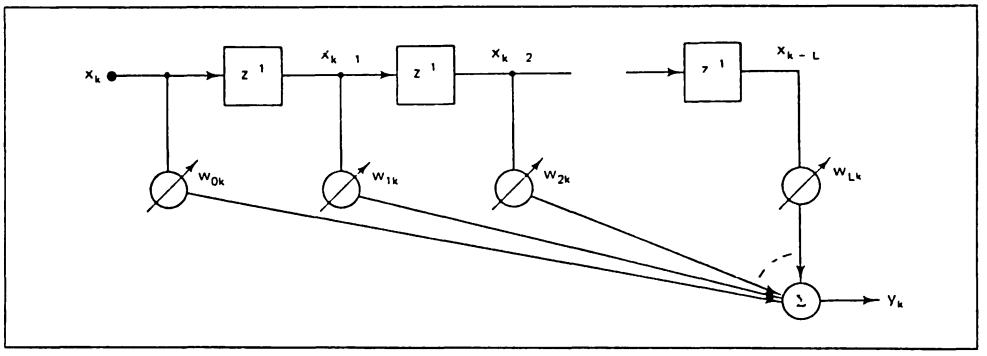


Figure 2.6.1 Single input adaptive linear combiner

$$X_k = \begin{bmatrix} x_k \\ x_{k-1} \\ \vdots \\ x_{k-L} \end{bmatrix} \tag{2.6.1}$$

$$W_k = \begin{bmatrix} w_{0k} \\ w_{1k} \\ \vdots \\ w_{Lk} \end{bmatrix}$$

The weight vector contains: (2 6.2)

The output is obtained from the following for a single input:

$$y_k = \sum_{l=0}^{L} w_{lk} x_{k-l} \tag{2.6.3}$$

This may be expressed in vector notation as:

$$y_k = X_k^T W_k = W_k^T X_k \tag{2.6.4}$$

As a simplified signal cancellation example; consider the case of a pure sinusoidal tone with frequency ω, amplitude A and sample period T used as the input to a single input, two weight FIR. The input is:

$$x_k = A\sin(\omega kT) \quad and \quad x_{k-1} = A\sin(\omega(k-1)T) \tag{2.6.5}$$

expanding gives:

$$x_{k-1} = A\sin(\omega \cdot k \cdot T)\cos(\omega \cdot T) - A\cos(\omega \cdot k \cdot T)\sin(\omega \cdot T) \tag{2.6.6}$$

The desired filter output is also a pure sinusoidal tone of the same frequency as the input with a phase shift ϕ and amplitude B:

$$y_k = B\sin(\omega kT + \phi) \tag{2.6.7}$$

Expanding gives:

$$y_k = B\sin(\omega \cdot k \cdot T)\cos(\phi) + B\cos(\omega \cdot k \cdot T)\sin(\phi) \tag{2.6.8}$$

Substituting the input into the FIR filter equation and solving for w_o and w_1 by equating coefficients of sines and cosines yields:

$$w_1 = \frac{-B\sin(\phi)}{A\sin(\omega \cdot T)} \quad and \quad w_0 = \frac{B}{A}\cos(\phi) + \frac{B\sin(\phi)\cos(\omega \cdot T)}{A\sin(\omega \cdot T)} \tag{2.6.9}$$

Notice that if we want y(k) = -x(k), this is easily achieved if ϕ = 180 degrees and B = A which corresponds to the coefficients w_0 = -1 and w_1 = 0. In general the phase angle ϕ and amplitude B are not simple to determine due to the transfer functions associated with the physical system and are different from the example above. The goal of the adaptive algorithm is to modify the weights such that the resulting gain and phase angle cancel the cabin noise. This example is very simple and does not consider the group delay, calculation delays and transport times. In reality several weights are necessary to account for all the delay and to control a tone.

2.6.2 Performance Function

Figure 2.6.3 shows a block diagram of a multiple input FIR filter with desired response and error signals. In the case of ANC the aim is to minimize the error signal. Each set of weights produces a corresponding error and a performance function is derived which relates the weights to the error. The performance function proves to be very important to the development of an ANC algorithm. The error signal will vary with time as:

$$\varepsilon_k = d_k - y_k \tag{2.6.10}$$

Substituting in for the filter output gives:

$$\varepsilon_k = d_k - X_k^T W_k \tag{2.6.11}$$

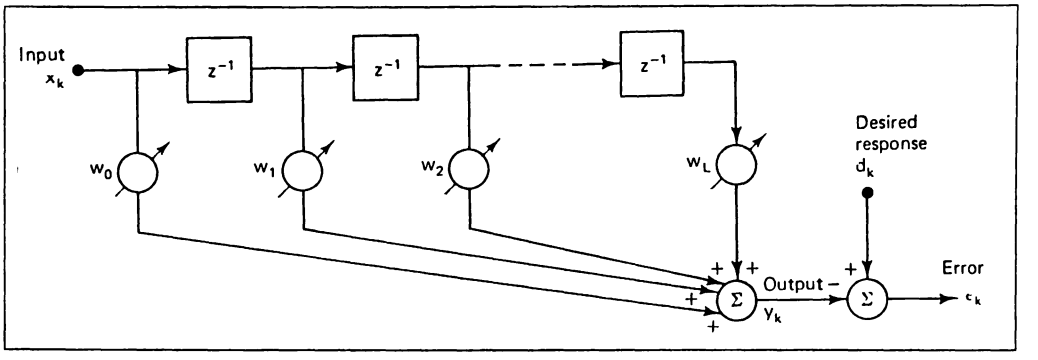


Figure 2.6.2 Adaptive linear combiner with single input, desired response and error

We are considering an instant in time when the weights are constant. We wish to evaluate the expected value of the square of the error signal at time, k, to determine the mean- square-error (MSE). First square both sides to obtain the instantaneous error.

$$\varepsilon_k^2 = d_k^2 + W_k^{\ T} \cdot X_k \cdot X_k^T W_k - 2d X_k^T W_k \tag{2.6.12}$$

Taking the error, desired and input data to be statistically stationary and taking the expected value at time k to determine the MSE. Recall that the expected value of a sum is the sum of the expected values and simplifying:

$$E[\varepsilon_k^2] = E[d_k^2] + W_k^T \ E[X_k \cdot X_k^T] \cdot W_k - 2E[d_k \ X_k^T] \cdot W_k \tag{2.6.13}$$

Recalling that the expected value of a product is the product of the expected values for independent variables allows further simplification. The input and desired variables can not be considered independent. Rewriting the MSE, ξ, as follows:

$$MSE = \xi = E[\varepsilon_k^2] = E[d_k^2] + W^T \cdot R_x \cdot W - 2P^T W \tag{2.6.14}$$

where:

$$R = E\left[X_k X_k^T\right] = \begin{bmatrix} x_{ok}^2 & x_{0k}x_{1k} & \cdots & x_{0k}x_{Lk} \\ x_{1k}x_{0k} & x_{1k}^2 & \cdots & x_{1k}x_{Lk} \\ \vdots & \vdots & \ddots & \vdots \\ x_{Lk}x_{0k} & x_{Lk}x_{1k} & \cdots & x_{Lk}^2 \end{bmatrix} \tag{2.6.15}$$

$$P = E\left[d_k X_k^T\right] = \begin{bmatrix} d_k x_{0k} \\ d_k x_{1k} \\ \vdots \\ d_k x_{Lk} \end{bmatrix} \tag{2.6.16}$$

R is the input correlation matrix where the diagonal terms are the mean squares of the input components and the remaining terms are the cross correlation amongst the input components. The P vector is the cross correlation between the desired response and the input components. The above is derived for a multiple input filter but a single input filter can also be used and similar results are obtained. Rewriting the MSE in terms of the correlation matrices and the weights clearly shows that it is a quadratic function of the weight vector when the input vector and the desired response are both stationary over time.

The minimum MSE is obtained with the optimum weight vector - also called the Wiener solution. The parameters of the performance surface are unknown and a closed form solution is not available when sampled data is used. Since the weight solution corresponding to the minimal mean-square error is desired, and only sampled data is available, an algorithm is needed which is capable of searching the performance surface to determine the optimal weight solution. In practice a near optimal weight vector is obtained in many algorithms by testing representative weight values and using the resulting mean-square error to modify the weight vector in a gradient search method. The performance surface gradient ($\nabla(\xi)$) simplifies to:

$$\nabla(\xi) = 2R_k W_k - 2P_k \tag{2.6.17}$$

The minimum error, ξ_{min}, is achieved when the gradient is zero since the performance function is quadratic. The weight vector corresponding to the minimal error, W* to evaluate the Wiener weight solution, is easily obtained by setting the gradient to zero.

$$W_k^* = R_k^{-1} P_k \tag{2.6.18}$$

The minimal error is obtained through the use of the Wiener weight solution and reduces to:

$$\xi_{\min} = E\left[d_k^2\right] - P_k W_k^* \tag{2.6.19}$$

Notice from the last two equations that both the Wiener weight solution and the minimum MSE are both dependent on the cross correlation between the input (reference) data and the desired (cabin noise in our case) data. This result shows that in order to apply ANC through the use of a FIR filter structure, the reference signal must cohere as closely as possible to the cabin noise to obtain large noise reductions. This observation proves to be very important to the practical implementation of noise control system and coherence requirements between the reference and cabin noise are discussed in detail in the remainder of this report.

2.6.3 Single Input, Single Output, LMS Algorithm

The Least Mean Squares (LMS) algorithm is applied to FIR filter structures and as an efficient method to obtain the gradient estimate needed to adapt the weights. Estimating the gradient of the MSE by taking differences between short term averages of the square of the error signal is prone to estimation error and requires a relatively large amount of calculation. The LMS algorithm uses the square of the error signal as the estimate of the MSE. The previously defined error signal, ε, is as follows

$$\varepsilon_k = d_k - X_k^T W_k \tag{2.6.20}$$

This method allows a quick and easy gradient estimate at each iteration:

$$\hat{\nabla}_k = \begin{bmatrix} \frac{\partial \varepsilon_k^2}{\partial w_0} \\ \vdots \\ \frac{\partial \varepsilon_k^2}{\partial w_L} \end{bmatrix} = 2\varepsilon_k \begin{bmatrix} \frac{\partial \varepsilon_k}{\partial w_0} \\ \vdots \\ \frac{\partial \varepsilon_k}{\partial w_L} \end{bmatrix} = -2\varepsilon_k X_k \tag{2.6.21}$$

With this estimate of the gradient, a steepest descent type adaptive algorithm is constructed with gain constant, μ, to obtain the LMS algorithm. This simple, efficient algorithm does not require any squaring, averaging, matrix inversion or differentiation. This algorithm does not use weight vector perturbations so each update of the weight vector reduces the error (in the mean-square sense) in the direction of the global minimum.

$$W_{k+1} = W_k - \mu \hat{\nabla}_k = W_k - 2\mu\varepsilon X_k \tag{2.6.22}$$

For all adaptive algorithms, the convergence behavior, the converged performance and the stability are major concerns. It can be shown that the gradient estimate is an unbiased estimate of the true gradient and that the weight vector solution does converge onto the optimal, Wiener solution (Ref. 16) with the proper choice of gain constant. The gain constant governs both the stability and the rate of convergence. This algorithm is not implemented directly in ANC applications because of the transport delays and frequency response functions associated with the transducers and error sensors. The LMS algorithm is still used but the input to the LMS is modified to account for these influences.

2.6.4 Multiple Input, Multiple Output, Filtered x LMS Algorithm

Consider the block diagram shown in Figure 2.6.4 with a system consisting of N reference signals, L error sensors (microphones) and M secondary sources (speakers). An FIR filter model is employed to calculate the control signals sent to the secondary sources. In this case consider a multiple input, multiple output (MIMO) system where a FIR filter is used on each reference signal to generate the output sent to each secondary source. The weight vector with components $w_{m,n,i,k}$ is read as the vector of length I which maps the n'th reference signal onto the m'th secondary source at time k and is used to generate the control signal $Y_m(f)$ as shown below.

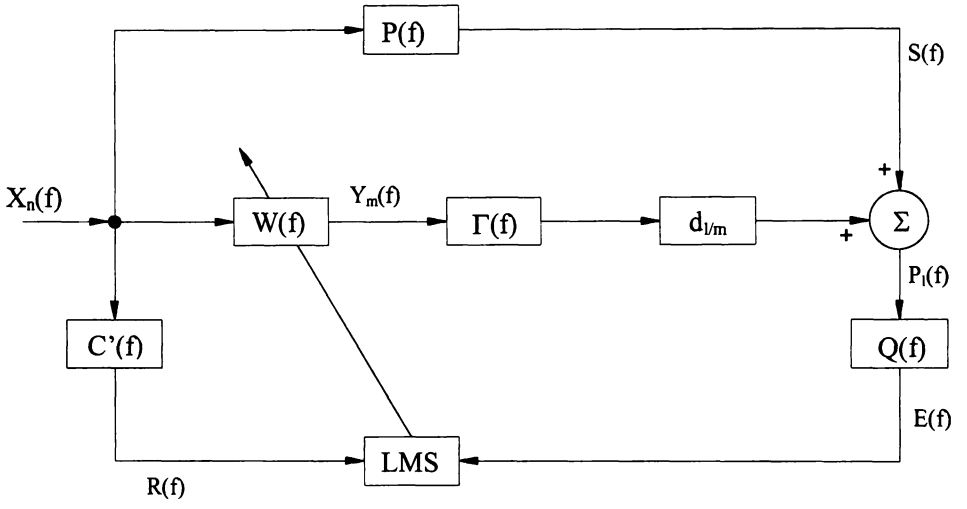


Figure 2.6.3 ANC system block diagram showing secondary path transfer functions and delay

$$y_{m,k} = \sum_{n=1}^{N} \sum_{\iota}^{I-1} w_{m,n,\iota,k} \cdot x_{n,k-\iota} \tag{2.6.23}$$

The acoustic pressure at each error sensor $p_{l,k}$ is the sum of the primary disturbance $S_{l,k}$ and the disturbance due to each of the control signals modified by the transfer function associated with each of the secondary sources. Each of the control signals is sent to its respective secondary source (speaker) and is modified by the electro-acoustic response function associated with each speaker, $\Gamma_{l,m}$. In addition to the response of the secondary sources, the electro-acoustic response of the feedback sensors (microphones) is taken into account through another FIR filter model, Q_l. The signal arriving at each of the microphones is not only different from the signal sent to the secondary sources because of these transfer functions, but is also delayed by dl/m samples due to the separation distance between the sources and microphones. This transport delay is assumed to constant, $d_{l/m}=d$, between all the secondary sources and microphones to simplify the notation and

derivation. The pressure at each microphone is written to include these additional terms (where * denotes the convolution):

$$p_{l,k} = s_{l,k} + \sum_{m=1}^{M} y_{l,m,k} * \Gamma_{l,m} = s_{l,k} + \sum_{m=1}^{M}\sum_{\iota=0}^{I-1}\sum_{n=1}^{N} w_{m,n,\iota} \cdot x_{n,k-\iota-d} * \Gamma_{l,m} \tag{2.6.24}$$

$$e_{l,k} = p_{l,k} * Q_l = \left(s_{l,k} + \sum_{m=1}^{M}\sum_{\iota=0}^{I-1}\sum_{n=1}^{N} w_{m,n,\iota} \cdot x_{n,k-\iota-d} * \Gamma_{l,m} \right) * Q_l \tag{2.6.25}$$

Defining the filtered primary source signal, $g_{l,k}$ and the filtered reference source signal, $r_{l,m,n,k}$ as follows to simplify the error equation:

$$g_{l,k} = s_{l,k} * Q_l \tag{2.6.26}$$

$$r_{l,m,n,k} = x_{n,k-d} * \Gamma_{l,m} * Q_l \tag{2.6.27}$$

The error signal is now rewritten as:

$$e_{l,k} = g_{l,k} + \sum_{m=1}^{M}\sum_{\iota=0}^{I-1}\sum_{n=1}^{N} w_{m,n,\iota,k-d} \cdot r_{l,m,n,k-\iota} \tag{2.6.28}$$

The above error signal is used as the input to an LMS algorithm to obtain the weight vector update equation for the 'filtered x' LMS (fxLMS) algorithm using the same method as the LMS algorithm. The fxLMS update algorithm is:

$$w_{m,n,\iota,k+1} = w_{m,n,\iota,k} - \mu \sum_{l=1}^{L} e_{l,k} \cdot r_{l,m,n,k} \tag{2.6.29}$$

Notice the weight update equation uses the filtered reference signal. The filtered reference signal is obtained through the use of a FIR model of the secondary path response C'(f) in Figure 2.6.1. The secondary path model must be determined prior to starting the control algorithm. Starting the

system is also simple if all the weights are initialized to zero since the control system does not increase the error above the noise level initially in the cabin. This is preferable to guessing the weights because the noise level in the cabin is never increased above the uncontrolled level. Since there are no division operations, there is no danger of dividing by zero.

The weight update equation is a function of the reference signal filtered by the response path between each of the secondary sources and feedback error sensors. The total secondary path response is due to Γ, Q and the delay and a FIR model of this response path is necessary to calculate the filtered reference signal. A system identification phase is used to model this response path with a series of FIR filters, each filter models the path from each source to each microphone. The system identification is easily performed through the use of an LMS algorithm as detailed in the following section.

2.6.5 Secondary Path Identification

The fxLMS algorithm relies on a model of the secondary path response. The secondary path response between each speaker and each microphone is modeled through the use of FIR filters. This section contains a brief explanation of how the LMS algorithm is utilized to perform the secondary path response identifications.

The secondary path identification is performed either on-line or off-line. An on-line identification continuously performs the identification at the same time the control algorithm is running. Such a system can compensate for large changes in the secondary path response. The off-line identification is performed before the control system starts and is not performed again during control. On-line identification is used in systems where changes in the secondary path response are too large or fast for the controller to compensate for unless a better model of the secondary path is obtained. Since the transport delay is a primary contributor to the secondary path response, it is instructive to investigate changes in the transport delay.

The speed of sound is proportional to the square root of the absolute temperature. A simple analysis to evaluate the transport delay as a function of temperature and sampling rate is easily performed. As the aircraft climbs, the temperatures will change. The temperature changes can be

very significant from over a sweltering 100 °F on the ramp to a comfortable 70 °F during flight or lower. If the system identification is performed off-line before takeoff with a 1,400 Hz sampling frequency and the temperature drops during flight to 70 °F, the transport delay increases. The change in the transport delay will effect higher frequencies more than lower frequencies. If the upper frequency is taken as 400 Hz the change in transport delay to travel 4 ft is only 4% of the period for a 400 Hz tone. This means the phase error is approximately 14 degrees and the difference in transport delay is less than a quarter of the sample period. Lower frequencies have even less error. The controller is expected to compensate for temperature changes of this magnitude and off-line identification is used in this work. On-line identification may prove to be necessary if future work shows the secondary path changes too much due to temperature or altitude changes for the controller to compensate. The effects of people moving around in their seats or in the cabin are not considered.

The secondary path responses includes the total response between the instant the control data is sent out by the controller and the instant it is received back by the algorithm for processing. The secondary path consists of the transfer functions and delays due to the A/D board, filters, amplifiers, speakers, microphones, the input filters and the A/D board. Let the FIR model of this path be denoted by C'. In the block diagram in Figure 2.6.5 the control weight vector, W, passes the input straight through without modifying the data. This is accomplished by setting the first weight value in W to one and the remainder of the weights are set to zero.

A reference signal composed of stationary, zero mean white noise which is uncorrellated with the cabin noise is fed to the controller. The identification can be performed during flight since the identification signal is not correlated with the cabin noise. The controller sends the reference data directly to the speaker without modification. The system is adaptively modeled through the use of the LMS algorithm until the identification is complete. The identification is terminated when the calculated MSE is some small fraction of the mean-square of the input signal (MSX).

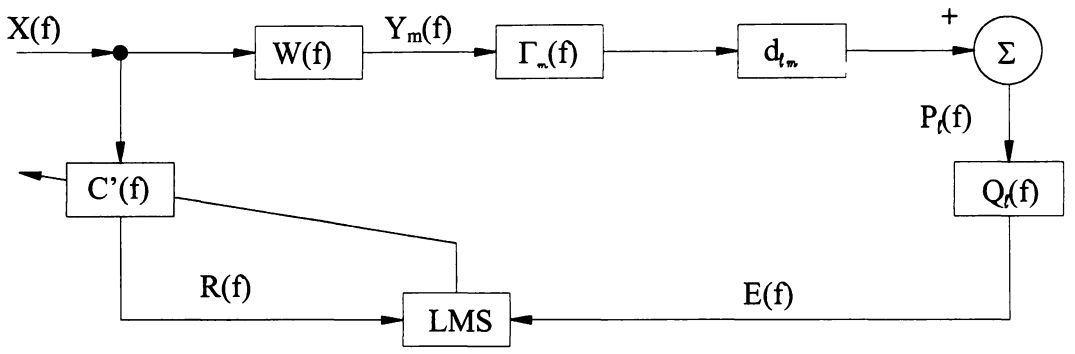


Figure 2.6.4 Block diagram of LMS algorithm implementation to perform secondary path identification

The identification of a system with a single speaker and microphone is easy but a system with more speakers and microphones is more involved because the output from each speaker is detected by each microphone. The speakers can all be identified simultaneously when a different white noise source is sent to each speaker so that the identification signals are uncorrelated to each other. It is easier to monitor the identification process when it is performed sequentially by only sending the noise to one speaker at a time. Noise is "pumped" through the first speaker and the identification is performed for the paths associated with that speaker and each of the microphones until the predetermined MSX/MSE is reached for each path. Once all these paths are modeled, the identification is terminated and the weights are stored. The process is then repeated sequentially with all the remaining speakers. All the resulting weights are stored for subsequent use by the fxLMS algorithm.

This section provides only a brief introduction to digital filtering and ANC algorithms. Convergence, stability and minimal errors associated with practical implementations are not discussed although there are several papers dealing with these subjects as referenced in the Appendix. It is instructive to perform some math modeling to gain some firsthand understanding

of the algorithm under different situations. The next section discusses some of the math models used and the lessons learned from the models.

2.7 MATH MODELING

Numerous papers dealing with the LMS and fxLMS algorithms discuss the performance, convergence, stability characteristics, and the effects of the transducer transfer functions and time delays are referenced. These references provide a large amount of information on the theory associated with the algorithm but it is desirable to obtain some firsthand experience with the algorithms in mathematical models before attempting any implementations in physical systems.

The goal of these models is not to completely simulate the system but rather to provide a simple learning tool to help determine the ideal performance before trying to control a real system. The math models are implemented in Mathcad (by MathSoft Inc.) which is used extensively throughout the research being reported as a convenient tool to perform quick testing of different scenarios.

Data to model the effects of the acoustics of the enclosure, the secondary path transfer functions and the primary path transfer functions are not included in the system models. The main reasons are that the data necessary to perform the modeling is not available and obtaining the data through experimentation is not the focus of this research. Collecting the data will shift the focus of the research and cause delays to the already lengthy duration of this project. The desired information can be obtained from much simpler models and most of the simulations are only for the SISO system.

2.7.1 Simulations

Delays and simple FIR models are used for the primary and secondary paths in most of the system simulations. Small filter lengths (8 to 20 weights) are used in most cases because only simple secondary path effects are included. A delay and a lowpass FIR filter are used to model the secondary path to test the system identification method. Most of the results are obtained by using simple data constructed in Mathcad although cabin noise recordings (Cessna 172) are used in some cases. Mathcad is also used to determine frequency domain transfer functions from weight vectors.

The Appendix contains several of the Mathcad worksheets. This section discusses some of the modeling results.

Figure 2.7.1 shows the time domain results of a system identification simulation where the secondary path is modeled by a lowpass FIR filter with a 3 sample period delay. The identification is started after 30 samples and the figure illustrates the rapid convergence (identification) as is expected in this simulation. The system converges after approximately 30 iterations.

The spectrum shown in Figure 2.7.2 is simulated cabin noise consisting of three tones and random noise which is used in many of the simulations. This data is used in a SISO fxLMS simulation as both the reference signal and the cabin noise (after applying a small delay). The residual noise spectrum after applying control is also shown in the figure. The large reduction in the tones is clear, as are large reductions in the broadband content below 200 Hz. These results represent the ideal situation where the secondary path is closely identified and there is close to perfect coherence between the reference signal and the cabin noise.

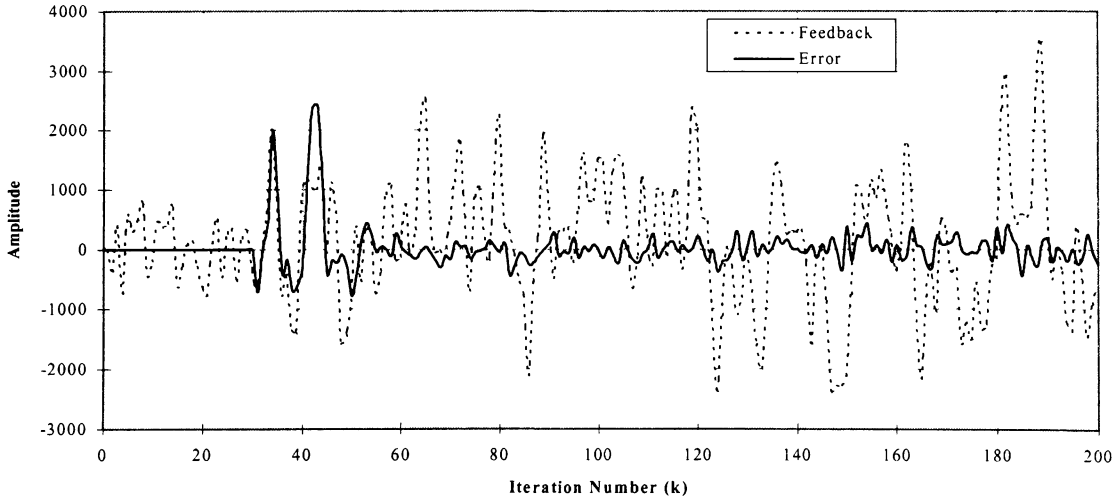


Figure 2.7.1 Time domain feedback and error data from system identification math model

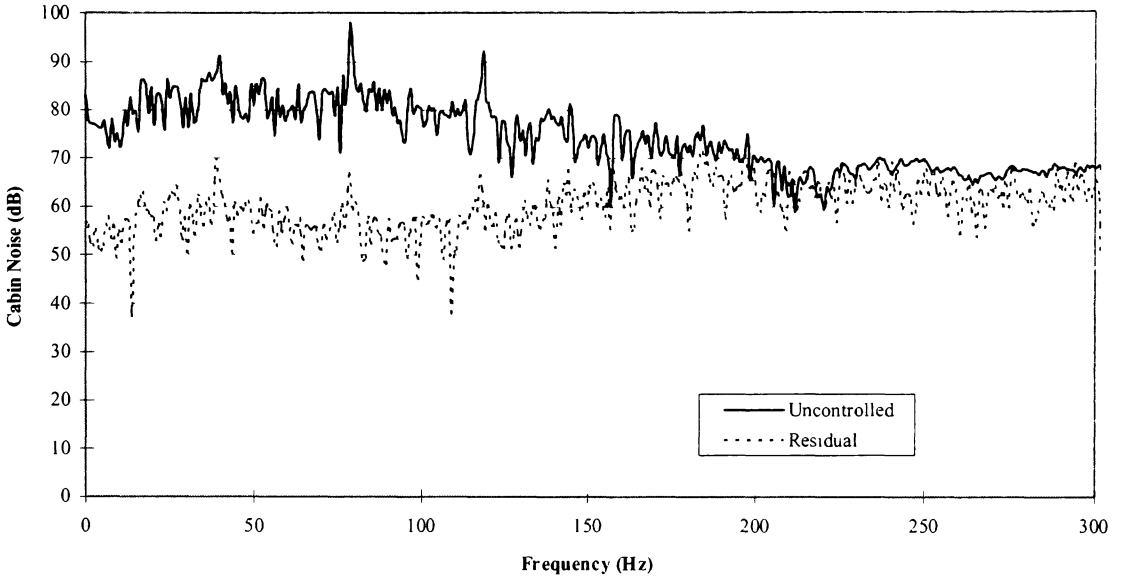


Figure 2.7.2 Simulated cabin noise spectrum before control and residual noise during control

SISO simulations to evaluate the performance when the reference signal does not cohere as closely with the cabin noise are presented. Reduced coherence can occur due to several reasons that depend largely on the method used to generate the reference signal. Reference signal generation is possible by several methods and each must be considered to ensure the best method is used. The type of control desired (broadband or tonal) also influences the feedforward method used. Broadband control requires a similarly broadband reference signal with strong coherence at all frequencies. Accelerometers, strain gages and microphones are examples of transducers which provide broadband signals. The transducer selection and placement determines the coherence achieved with these methods. Tonal control is also possible with these types of feedforward transducers. The reference signal section of this chapter contains more details.

Figure 2.7.3 shows the tonal reference and noise signals constructed in Mathcad. There are two tones in the noise signal which are absent from the reference signal which means there is very weak coherence between the reference and noise signals at these frequencies. The reference signal is constructed to contain just two tones and has no spectral content other than these two frequencies. The figure shows the reference signal to have spectral content at other frequencies but this is

calculation noise due to the 1024 point Fast Fourier Transform calculation which is used to evaluate the spectrum. There is close coherence between the tones in the reference signal and the corresponding tones in the noise signal.

Figure 2.7.4 illustrates the feedback signal before control and the residual noise when control is implemented. By comparison of the two figures, notice those tones showing weak coherence are not reduced by any significant amount and the tones which show strong coherence are reduced. This simple example illustrates the importance of coherence between the reference and feedback signals. If the reference signal has poor coherence with the noise, the use of more than one reference signal can result in larger reductions if additional reference signals improve the coherence.

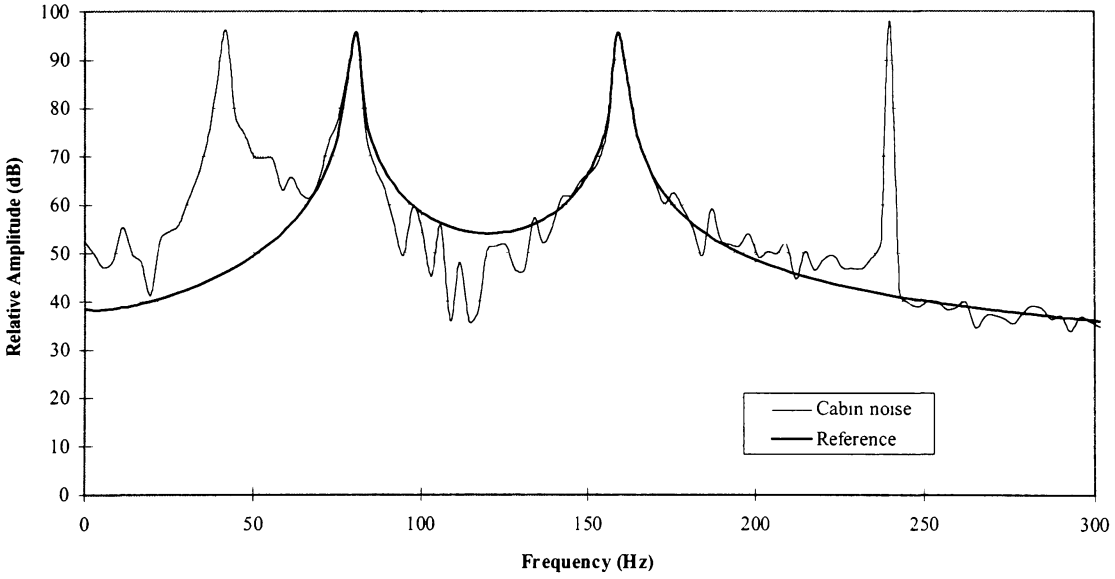


Figure 2.7.3 Simulated cabin noise and tonal reference signal

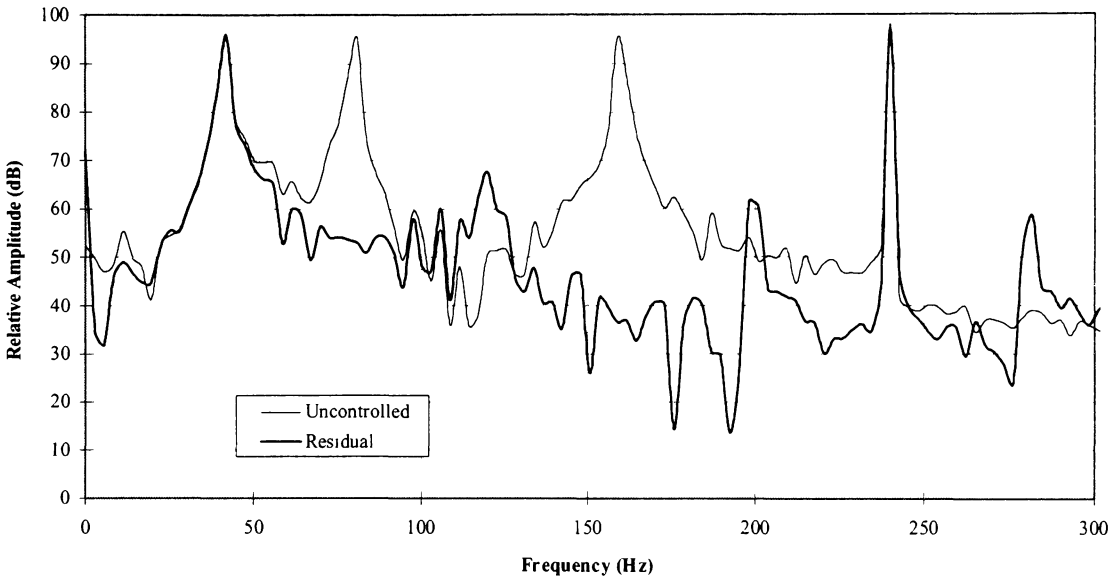


Figure 2.7.4 Simulated cabin noise before control and residual noise during control

The use of two reference signals is tested using a similar noise signal as the previous simulation and two reference signals as shown in Figure 2.7.5. Notice that the ratio of the amplitudes of the tones in the reference signal is the same as the ratio for the corresponding tones in the noise signal. These ratios indicate strong coherence between the reference signals and the cabin noise. At frequencies where one of the feedforward signals does not cohere with the cabin noise, the other signal does. Figure 2.7.6 shows uncontrolled and the residual noise spectra. Large reductions at each of the dominant tones are once again achieved because of the good coherence. These results also show the expected results that when tonal reference signals are used the broadband components can not be reduced. These simulations illustrate the importance of the reference signal coherence with the cabin noise.

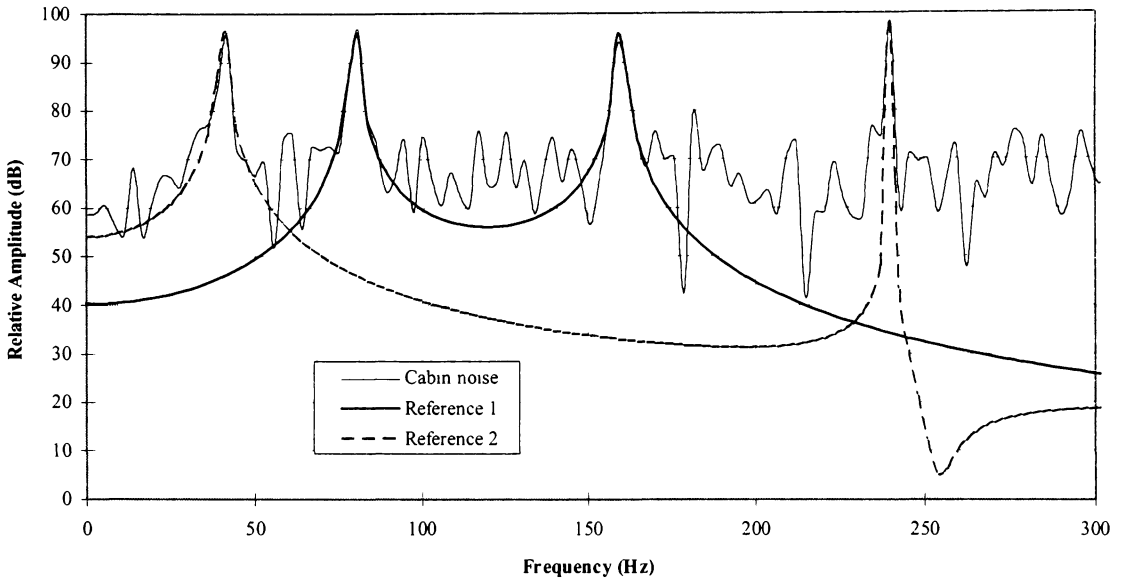


Figure 2.7.5 Simulated cabin noise and two tonal reference signals

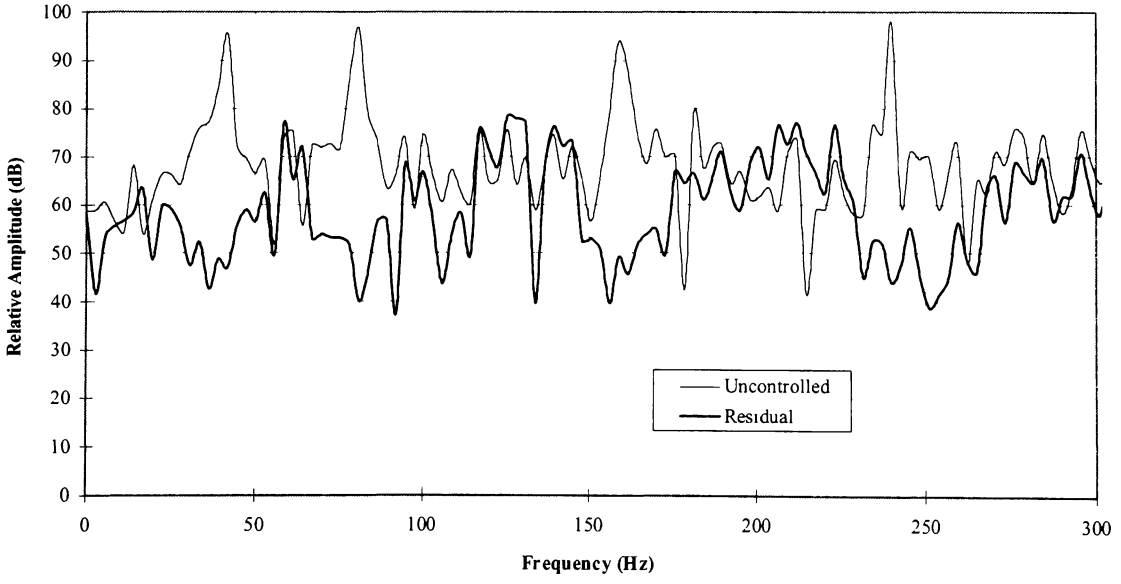


Figure 2.7.6 Simulated cabin noise before control and residual noise during control

2.7.2 "A" Weighted Sound Spectrum

The human ear is the least sensitive to low frequency sound and is most sensitive to sound around 1 kHz. "A" weighted and "C" weighted overall-sound-pressure-levels (OASPL) are obtained by modifying the sound level at each frequency to correspond to the human response to soft and loud sounds respectively. The weighting attempts to account for the difference between the perceived sound levels and the actual sound levels which has the effect of modifying the spectrum. ANC system performance is often quoted in terms of the reduction in "A" or "C" weighting sound levels for these reasons. The effects of applying "A" weighting to the cabin spectrum is investigated to determine if it is possible to improve the perceived noise reductions.

Figure 2.7.7 shows the "A" weighting relative response spectrum which is a model of the sensitivity of the human ear to loud sounds of different frequencies (Ref. 11). Notice the effect of "A" weighting at frequencies up to about 600 Hz is to reduce the intensity in a very nonlinear relationship with frequency. At 80 Hz (the vicinity of the loudest BPF tone in cabin noise), the intensity is reduced by approximately 23 dB from the original value. The effect of "C" weighting is to reduce the contribution to the OASPL by the BPF tone and increase the contributions by the BPF harmonics. The "C" weighted response does not alter the low frequencies as extensively as "A" weighted response which is why this discussion is limited to "A" weighing only.

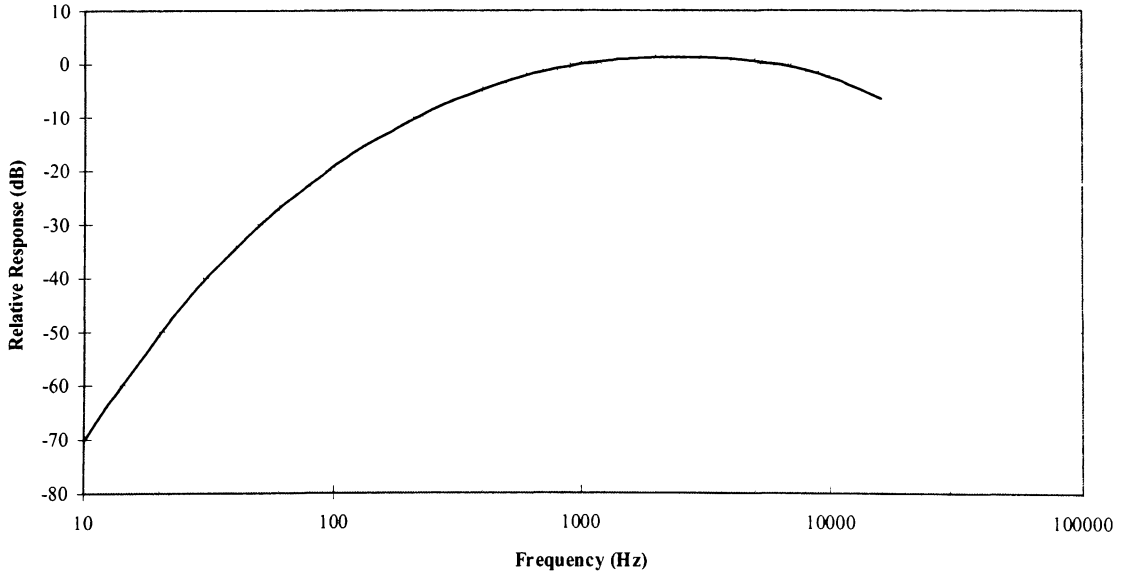


Figure 2.7.7 "A" weighted relative response spectrum

Implementing the system with data received directly from the microphones results in the maximum reduction in unweighted sound pressure level decibels. The possibility exists that applying "A" weighting to the uncontrolled spectrum causes the BPF harmonics to become just as dominant as, or more dominant than, the BPF tone itself. If the system performance is evaluated with A weighting and the BPF harmonics become equally as important as the BPF tone, it is advantageous to implement an A weighted algorithm. Brief consideration is given to applying "A" weighting to the data through the use of FIR filters or "A" weighting circuitry before using the data in the algorithms.

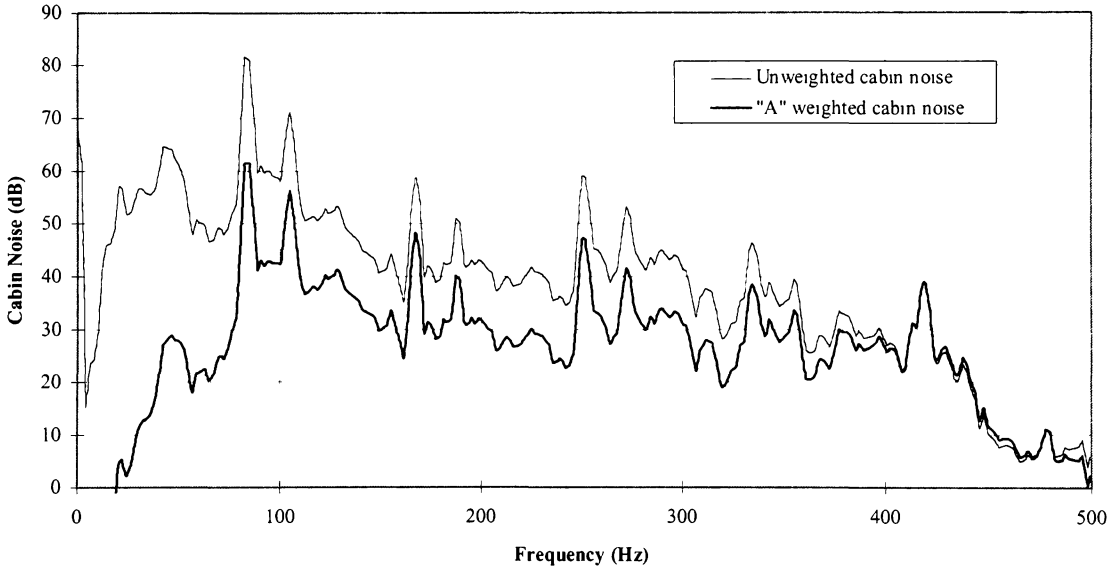


Figure 2.7.8 Resulting spectrum after applying "A" weighting relative response spectrum to Cessna 172 cabin noise recorded during cruising flight

Figure 2.7.8 illustrates the Cessna 172 cabin noise spectrum before and after applying the "A" weighting response spectra to the cabin noise. The "A" weighted spectrum shows the BPF tone to dominate the harmonics by approximately 10 dBA (compared to 20 dB unweighted). The tone at approximately 110 Hz is only 5 dBA below the BPF tone compared to the original 10 dB difference between these two tones. Also note the low frequency components below BPF are reduced in amplitude by a very large amount. The first and second harmonics of BPF are only 10

dBA below the BPF amplitude compared to the original 20 dB difference. The effect of the "C" weighting is to raise the amplitudes of the BPF harmonics relative to the BPF amplitude.

The value of implementing any sort of weighting in this research is uncertain because the same tones are still dominant. If the resulting spectrum shows one, or some, of the higher frequency tones to become dominant over the BPF tone, then applying a weighted control system is useful. "A" and "C" weighting are not applied in this research.

2.8 REFERENCE SIGNAL CONSIDERATIONS

The previous sections illustrate the importance of strong coherence between the reference signal and the cabin noise. Several considerations are important to the selection of the reference signal source. For all feedforward methods, the primary concern is to ensure strong coherence with the cabin noise over the lifetime of the system.

Tonal noise control only requires the reference signal to cohere with the cabin noise at the targeted tones and a tonal reference signal can be used for this purpose. Frequency content other than the tones of interest is not desirable and should be minimized in tonal control. A tonal reference signal is easily generated by filtering a pulsed signal. The pulsed signal may be obtained by several methods with the basic idea being to obtain a signal with a pulse frequency corresponding to BPF. A pulse contains components at all frequencies which are integer multiples of the pulse frequency. The harmonics of BPF therefore correspond in frequency to the filtered components of the pulse. The pulsed signal can be generated by using a stationary magnetic pickup together with two magnetic source mounted on the propeller shaft (assuming a two bladed propeller). As the propeller shaft rotates, the magnetic sources trip the magnetic pickup twice per revolution. The number of magnetic sources must match the number of propeller blades if one magnetic pickup is used.

As an example consider an engine rotating a propeller shaft at 2,400 rpm which spins a propeller with 2 blades. BPF in this case is 80 Hz and using the above scheme, two pulses are generated per revolution of the propeller shaft which also corresponds to 80 Hz. After filtering the pulse train

with an appropriate lowpass filter to eliminate all but the lowest three frequencies, the resulting signal contains components at 80, 160 and 240 Hz which corresponds to BPF and the first two harmonics. However, filtering a pulse train produces tones which all have the same amplitude and the reference signal can not be used as is because it has poor coherence with the cabin noise at the harmonics. To improve the coherence, the lowpass filter must have a shaped response function or a FIR filter is applied before the reference signal is used by the algorithm. Such a system requires extensive work before it can be applied in different aircraft due to the different cabin spectra in each aircraft. The change in the structural-acoustic response of the aircraft as it ages can also cause coherence problems with this method. An alternative tonal reference method which is more versatile, simpler to implement and has improved performance potential is discussed next.

The system setup is similar in that the same method is used to generate a pulsed signal. In this case several bandpass filters are used and one reference signal is generated for each tone to be controlled. As many reference signals are used as there are cabin tones which are targeted for control. However, the use of filters to generate the reference signals is problematic due to the harmonic content of the filter output and the very high filter orders required. The high filter order can also present a problem in terms of frequency tracking. An improvement to using filters is to use the original pulse train and a series of integrated circuits (IC) to perform frequency multiplication and division to generate several pulse trains corresponding in frequency to each of the targeted tones. Phase locked loop voltage controlled oscillator IC's are used to convert each pulse train to a tone. The voltage controlled oscillators need to be phase locked to prevent phase drift over time with respect to the phase of the corresponding cabin tone. Several different IC's exist which can be adopted for this purpose.

It is preferable to use each tone as a separate reference signal because the weight vector acting on each tone is then updated independently of the other tones. The use of one tonal reference signal for each targeted tone effectively eliminates coherence problems as long as the reference tone has the same frequency as the cabin tone and the phase difference between them does not drift over time. If the tones are combined into one reference signal, the resulting spectrum must be shaped to obtain sufficient coherence with the cabin noise to enable control. Such spectral shaping through the use

of different gains or passband filtering is difficult to implement and the resulting system is not as versatile.

There are additional gains to be had by using multiple tonal reference signals. Recall the simple example which illustrates that only a few weights are needed to control the amplitude and phase of a pure tone. This means relatively few control weights are needed by the control algorithm which significantly reduces the number of calculations. The system identification is also simplified along similar lines through the use of narrowband system identification such that only a few weights are required to filter each of the reference signals. The benefit of minimal calculation must be balanced against the increased number of A/D channels required.

The drawback to noise control using a tonal reference signal is that only those tones present in the reference signal are reduced in the cabin. Engine tones, propeller imbalance and other tones which are not present in the reference signal can not be reduced and broadband reduction is not possible. Frequencies other than BPF and BPF harmonics can be constructed through the use of frequency multiplication and division but there are limits. As mentioned earlier, once the tonal components are reduced to the broadband level, any additional noise reductions must be broadband for the effects to be noticeable. An example of a system which achieves broadband reduction in addition to reducing the propeller tones is the Lord Corporation system for the King Air (Ref R1.1) which uses an accelerometer mounted in the nose compartment of the (twin engine) aircraft.

The attraction to using a broadband reference signal is that it is possible to use a single reference signal filtered by a single lowpass filter which is easily designed. By suitable transducer selection and placement, the necessary coherence is obtained to allow tonal control and in some implementations broadband control is possible. For this research it is desirable to minimize cost and modifications to the aircraft so using an accelerometer or strain gage mounted to the structure is avoided. This is especially important in this research since the intent is to collect data in different aircraft. The multiple, tonal reference method is also rejected for the same reasons.

Previous research shows strong coherence between windshield vibrations and cabin noise as reference earlier. The windshield vibrations drive the acoustic excitation of the cabin and for this

reason a single feedforward microphone, placed in close proximity to the front windshield, is selected as the method of choice. The math modeling indicates extensive broadband reductions are possible if there is sufficient coherence and this feedforward method will show if any broadband reductions are possible. With such an acoustic feedforward system coupled with the fxLMS algorithm, it is necessary to isolate the feedforward microphone from the sound produced by the secondary sources. This is discussed in the Microphone section elsewhere in this report.

2.9 SYSTEM CONFIGURATION

A simplified block diagram is provided in Figure 2.9.1 to illustrate how the different components relate to each other in the final system design which is implemented. The primary noise source is shown by the propeller and engine. The transfer function P(f) is used to indicate the total structural/acoustic response of the aircraft that contributes to the cabin noise as detected by the feedback microphones in the cabin. The feedforward microphone generates the reference signal which is filtered, amplified and converted to a digital value by the A/D board. The next chapter details the A/D and D/A boards and the filters.

The reference signal is used by the control algorithm to generate the controller output through the use of FIR filters as illustrated in the block diagram. The output is converted to an analog voltage by the D/A board, passed through a smoothing filter and amplified before reaching the secondary source speakers. The sound created by the speakers travels to the feedback microphones which generate proportional voltage signals. The feedback signals are filtered and converted to digital values which are passed to the LMS algorithm. The electro-acoustic transfer functions and delays associated with these signal paths are modeled by a series of FIR filters. During system identification secondary path FIR filters are constructed to model each transfer function, C'(f) in the block diagram. These filters are used to generate the filtered reference signals which are used in conjunction with the feedback signal by the LMS algorithm to update the controlling FIR weights.

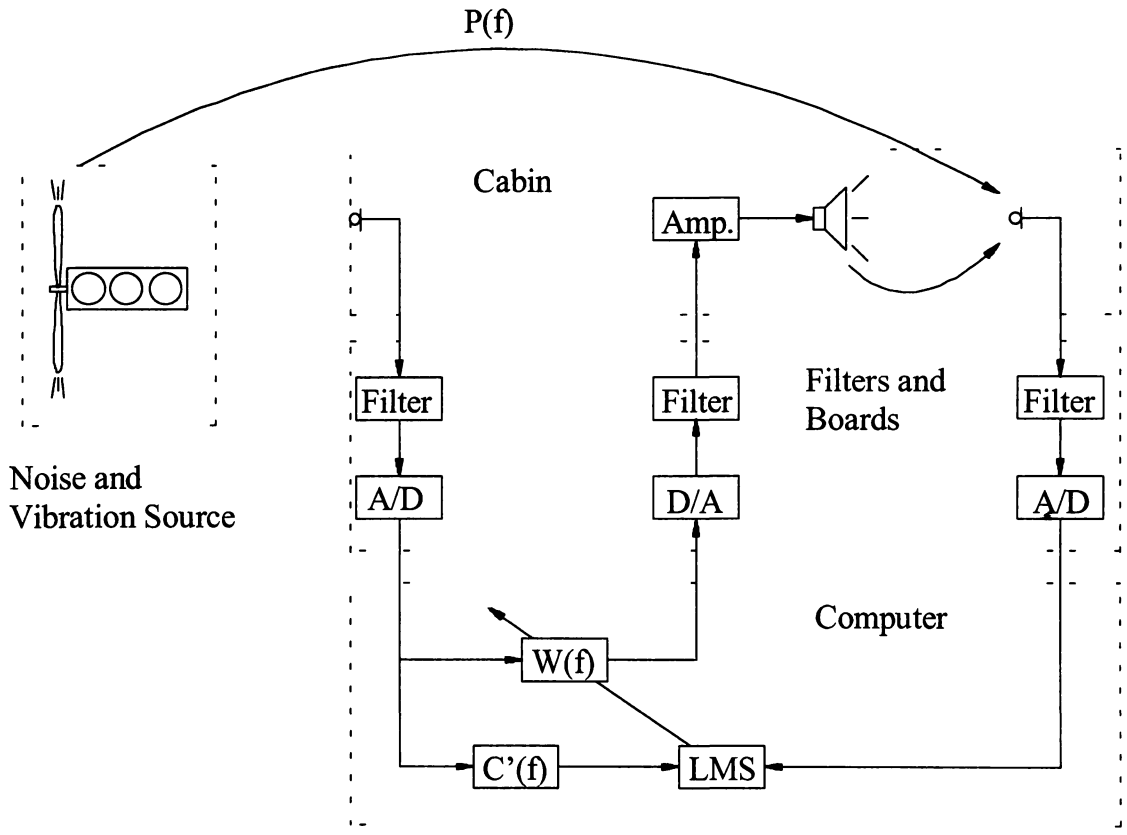


Figure 2.9.1 ANC System block diagram

A computer is used to implement the system identification and control algorithms discussed previously. The computer must also work with the A/D boards as outlined elsewhere. Analog filtering is necessary to limit the frequency range of the reference and feedback signals. Smoothing filters are necessary to convert the stepped voltage output from the D/A board to a smooth analog signal which is used to drive the secondary sources. The next chapter details the experimental hardware and software components, and the methods used to implement the control system.

CHAPTER 3
EXPERIMENTAL

The equipment, hardware development and the implementation of the algorithms are presented in this chapter. This chapter discusses the practical implementation of the theory discussed in the previous chapter.

3.1 BOARD CONTROL AND ALGORITHM PROGRAMMING

The algorithm and board control are implemented in 'C' coding on a 100 MHz Pentium, notebook computer. The notebook computer is installed on a docking station to allow the use of a digital-to-analog conversion (D/A) board and an analog-to-digital conversion (A/D) board. As mentioned previously, 12 bit A/D and D/A converter boards are used. Appendix B lists the make and model of each component. This section outlines the control of the boards and the practical implementation of the algorithms.

3.1.1 Board Control

An A/D board is required to convert the reference and feedback signals to digital data. The resulting digital data is imported into the computer where the control algorithm uses the data to generate the control signals and update the control weights. The control signal generated by the computer is exported to the D/A board where the digital control signal is converted to a voltage level which is then passed through the smoothing filters before it reaches the amplifiers. Both boards operate at the same fixed sample rate which is discussed in more detail in the next section.

Since the control filter is in the form of a digital FIR filter, data must continuously pass through the computer, even after the system has converged to a minimal error state. This means control signal updates can only be sent out by the computer after all necessary calculations are complete. These real-time requirements demand some consideration to ensure the algorithms and board control are implemented in a manner that is efficient at runtime. Ideally the boards are running as a background operation while the computer is processing the data through the algorithm. When the computer has completed all the necessary calculations, new data is imported from the A/D board and new control data is exported to the D/A board.

The exchange of information between the boards and the computer can be controlled by either interrupt programming or direct memory access (DMA) programming. Interrupt programming is a synchronous method where the CPU is caused to suspend its current task to service the designated activity which generated the interrupt signal. An example of interrupt level programming is when a command is issued which causes the A/D board to collect a sample. The CPU is suspended and waits until the A/D board returns the requested data and releases the interrupt, at which time the CPU continues its activities. This method is not very efficient because the CPU cannot process the algorithm while data collection is in progress.

The delay associated with serving the board though interrupt programming is significant because the software must issue several control bytes detailing the sample duration for each channel to be sampled, the vector of channel numbers and the vector of gain channels. After each control byte is written, the computer waits until a response is received from the board, usually in the form of status byte. The board then initiates the data conversion and the CPU must wait until the data conversion is complete. Several more exchanges take place between the board and the computer before control is returned to the CPU. A more efficient method to exchange information with the boards is through the use of DMA programming.

When DMA access is used, the boards are first initialized. During initialization, memory is allocated in the computer which the boards use to transfer data to and from the computer. The board runs as a background operation independent of the CPU activity. When data exchange is needed between the computer and the board, a DMA request is served to the computer's DMA controller which allows data to be passed through the DMA channels to, or from, the allocated memory locations as a background operation without interrupting the CPU. Both the boards and the CPU can initiate DMA requests. The CPU interruptions associated with DMA requests are several orders of magnitude less than those due to interrupt programming. DMA is used in this research to maximize the amount of time available to the CPU to perform the necessary algorithm calculations.

Before starting the control algorithm, the boards and computer are configured. The A/D board is configured and controlled through the use of software only. The D/A board requires jumpers and dip switches to specify some of the board parameters and the remainder of the setup is software controlled The board set up and control is not detailed because it is very specific to the make and

model of each board. The board initialization and control performed through the use of software is briefly outlined below.

Control bytes are written to specific memory addresses in the computer that are accessed by the boards. Each bit in the control byte serves a different function. After each control byte is issued, the computer must wait until a status byte is written to the computer by the board. The status byte is used to indicate the board status, the status of an operation on the board or any board errors. After the status byte is received the computer can proceed if there are no errors. Register level programming through the use of the 'C' language is used to implement control. The boards include driver software in the form of 'C' libraries . The driver software is optimized by the manufacturer and greatly simplifies the programming task.

Setup consists of specifying the channels used on each board and the voltage range, gain, sample duration and sample rate associated with each channel. Memory addresses are specified in the computer for the A/D board to write data to and the D/A board reads data from another series of memory addresses. The 'C' code listings in Appendix C is an early version of the system identification and noise control programs. These programs perform all board setup, initialization and control through the use of the driver software. The code is shown only as an example and will only function with the systems detailed in this report.

3.1.2 Algorithm Programming And Circular Buffers

The derivation of the LMS algorithm shows the FIR filter requires input data containing the most recent, sampled, data point and a number of sequentially older data points to perform the convolution with the FIR weights. The derivation shows the input vector with the data arranged in a chronological order with the most recently sampled data at the top of the vector and the oldest sample at the bottom of the vector. To implement the algorithm in this manner is very inefficient because it requires all the data in the vector to be shifted down one position to drop the oldest data point and make room for the newest data point. In the computer, data is not manipulated as shown in the matrix format used in the LMS derivation but through convolutions as shown in the fxLMS derivation. Circular buffers are used to implement the algorithm in an efficient code which requires minimal copying or memory shift operations.

The software circular buffers are arrays in memory which are constructed to contain the raw input and the filtered input data. The term circular buffer is applied because of the way these arrays are used. In the array the newest data point is written to the array position occupied by the oldest data point which effectively replaces the oldest data point with the newest. This means that all the other data points do not need to be copied to another position in the array every time a new data point is inserted. A pointer is used to track the location in the array occupied by the newest data point. To use the array in a convolution, pointer arithmetic is used to determine the position of the next data point in the circular buffer. If the next data point lies beyond the end of the array, the pointer needs to wrap around to point to the beginning of the array.

Whenever analog time varying data is sampled to obtain a digital representation, aliasing is a concern. Frequencies in the analog spectrum above the Nyquist frequency are folded onto lower frequencies, corrupting the spectral content of the digital data. Analog anti-aliasing filters are used to minimize aliased data. The digital output is changed to a voltage by the D/A board and this voltage value is maintained by the board until a new voltage is written to the board. The resulting analog output is not usable and is filtered to smooth the transitions between the voltage updates. The design and implementation of the anti-aliasing filters and smoothing filters is discussed in the next section.

3.2 FILTERS

Analog filters are used to condition the signals going into the A/D board and coming out of the D/A board. Anti-aliasing filters are required to prevent aliasing the input data. The frequency range in which ANC is known to be effective is much smaller than the total frequency range of the acoustic spectrum so the feedforward and feedback signals must be filtered to eliminate frequencies above those of interest and to eliminate dc components. Smoothing filters are implemented on the outputs because the D/A board operates on an update-and-hold scheme.

3.2.1 Anti-Aliasing Filters

The sampling frequency must satisfy the Nyquist criterion (Ref. 18) which requires the sampling frequency to be greater than twice the highest frequency contained in the input signal to prevent aliasing. Lowpass anti-aliasing filters are necessary to filter the input signal to each A/D converter

to attenuate those frequencies greater than the Nyquist frequency to prevent aliased data. Frequencies higher than the Nyquist frequency must be attenuated to amplitudes so low as to be undetectable by the A/D converter. The anti-aliasing filter requirements are determined through the use of Reference 19 and is outlined below. The number of bits used by the A/D converter to digitize the analog signal determines the resolution. The larger the number of bits, the higher the resolution and the smaller the detectable voltage change. The voltage range (V_{ref}) between the minimum and the maximum voltage the converter can span and the number of bits (n) determine the voltage resolution (q) as follows:

$$q = \frac{V_{ref}}{2^n} \tag{3.2.1}$$

The stopband frequencies must be attenuated to less than the RMS quantization noise level of the converter which is calculated as follows:

$$V_{qnoiselevel} = \frac{q}{2\sqrt{3}} \tag{3.2.2}$$

The amount of attenuated is calculated in dB as Q_{rms}:

$$Q_{rms} = -20 \log_{10} \frac{V_{ref}}{V_{qnoiselevel}} = -20 \log_{10} 2^n 2\sqrt{3} \tag{3.2.3}$$

For a 12 bit converter, approximately 83 dB of attenuation is needed in the stopband. Filter rolloff is approximated at 6N dB per octave where N is the order of the filter. The filter order needed to obtain this attenuation depends on the sample rate (f_s) and the corner frequency (f_c). For f_c close to the Nyquist frequency ($f_N = f_s/2$), a very high filter order is needed because of the very steep rolloff required to obtain the necessary attenuation at f_N. A conservative estimate of the filter order is obtained from:

$$N = \frac{|Q_{rms}|}{6 \log_2 \frac{f_N}{f_c}} \qquad where \quad \log_2 \frac{f_N}{f_c} = \frac{\ln \frac{f_N}{f_c}}{\ln 2} \tag{3.2.4}$$

A sampling frequency 5 times greater than the upper frequency of interest satisfies the Nyquist criterion and produces the desired acoustic clarity when the data is converted back to an analog signal. Since the upper frequency of interest is 300 Hz, the sampling frequency must be approximately 1,500 Hz (5 x 300 Hz). From the cabin noise discussion, the highest BPF harmonic of interest is expected to be the second harmonic. At a BPF of 90 Hz (2 bladed propeller at 2,700 rpm), the second harmonic is at (3 BPF) 270 Hz. Based on these considerations a 1,400 sample per second sampling frequency is used.

To obtain the full attenuation calculated above with f_s = 1400 Hz and f_c = 400 Hz requires (N=18) and 18 th. order filter which is not feasible. After examining the available cabin noise spectra for the Saratoga and Cessna 172, it can be seen that at 300 Hz the amplitude is approximately 40 dB below the BPF amplitude. The voltage range is largely determined by the amplitude of the BPF and harmonic tones. Since the main interest is in these tones, the attenuation requirements are adjust for this interest such that Qrms = -(83-40) = -43 dB and 300 Hz is used as the corner frequency. These inputs result in a 6th order filter requirement which is still a high order filter but is made possible through the use of commercially available integrated circuits (IC's). Some aliasing occurs but this appears as low amplitude, uncorrellated white noise, mostly at the lower frequencies where the tones are the loudest and so the aliasing will have the least effect at these frequencies.

Oversampling is a technique where the analog filter requirements are reduced because a much faster sampling rate is used which allows a larger transition between the corner frequency and the stopband frequency. Four times oversampling means sampling at 4 times the current rate (f_s = 4 x 1,400 Hz = 5,600 Hz) and only requires a second order filter (f_c = 300 Hz, Q_{rms} = -43 dB). Such a scheme produces 4 times as many data points and it is possible to use only every 4 th. data point by employing a decimation scheme so that the amount of data processed is the same. However, since only frequencies up to 300 Hz are desirable in this application, a decimation algorithm which uses only every 4th sample must pass the data through a lowpass FIR filter. Implementing 4 times

oversampling and a decimation scheme which contains such an FIR filter is easily implemented but increases the amount of calculations needed and oversampling is rejected at this stage in the research.

Two sixth order filters are constructed, each with variable gain and an adjustable corner frequency (approximately 250 Hz to 400 Hz). The filters are constructed through the use of model LMF60-50 High Performance 6th-order switched capacitor Butterworth lowpass filter IC's by National Semiconductor. Appendix D contains a circuit diagram of one filter implemented with a single-sided (battery) power supply. This IC is a switched capacitor type filter so care must be taken to prevent aliasing in the filter itself. A single order lowpass RC filter is implemented to prevent aliasing of the filter and this has the added benefit of providing additional filtering above 700 Hz. Switched capacitor filters provide update-and-hold output necessitating a lowpass smoothing filter which is implemented on the filter output. A single order highpass RC filter is implemented on each filter input with a 20 Hz corner frequency. The highpass filter is used to reduce large amplitude, very low frequency components because these frequencies are not audible and can saturate the A/D converters. Another single order highpass RC filter is implemented in the power supply.

3.2.2 Smoothing Filters

The D/A board updates the voltage level for each of the selected channels and holds that voltage until a new value is written to it. The voltage at such an output channel appears as a series of voltage steps. The sampling rate is sufficiently slow that the update rate is audible and dominates over the actual signal of interest. A step signal consists of many tones whose frequencies and amplitudes are related to the step frequency and amplitude. The amplitude of these tones are so large (compared to the actual signal being updated) that the signal of interest is not audible unless a lowpass smoothing filter is used to smooth the steps between updates. Two Krohn-Hite model 3340 laboratory filters are used for this purpose (one for each channel of output). These 6th. order filters are used with corner frequencies ranging from 250 Hz to 600 Hz.

3.2.3 Precision Voltage Reference

The D/A board reference voltage can be adjusted through the use of a combination of hardware and software. The board is jumper configured for bipolar input (as is the input board) with an output voltage range of approximately -1.2V to +1.2V. A model LM336 2.5V National

Semiconductor precision voltage regulator is used for this purpose, powered by the same battery supply as used to power the anti-aliasing filters. The reference voltage is applied to the appropriate pin on the board and the reference voltage value is set in the software during the board initialization.

3.3 MICROPHONES

Microphones are used for both the feedforward system (reference signal) and the feedback system (feedback signal). Although a highly accurate microphone is not necessary for this research because it is part of the system which is identified by the algorithm and the system will try to minimize the microphone signal, it is still desirable to use a microphone which has a flat response. Ideally the microphone response and signal-to-noise ratio should be matched to the signal conditioning filters and the A/D board characteristics to make the best possible use of all available voltage resolution. The anti-aliasing filters have adjustable gain for this purpose.

Panasonic Electret condenser type microphone cartridges are used for both the feedforward and the feedback. These microphones are intended for interim use before microphones of better quality are acquired. After testing the microphones and achieving positive results, these microphones are retained for the duration of this project due to time constraints and the preliminary nature of this work.

It is desirable to check the performance of the microphones by comparing the performance with a microphone of established quality. For this test a B&K model 2230 Precision Integrated Sound Level Meter microphone is utilized. The condenser element of the microphone is placed next to that of the B&K microphone. The two condensers are placed such that their open ends are parallel to each other and almost touching. The object is to excite both microphones with identical acoustic spectra and compare the voltage outputs of the Electret microphone to the B&K microphone. The output of each microphone is fed to the input of a Hewlett Packard 3582A spectrum analyzer. Bandwidth limited white noise is used to drive a speaker, and hence the microphones, and the output spectrum for each microphone is stored on the spectrum analyzer. The RMS average of 4 ensembles is recorded on the spectrum analyzer. The Electret microphone is used in conjunction with the filters and power supply which are discussed elsewhere.

Figure 3.3.1 shows the spectral results of the comparative test performed to evaluate the performance of the microphones. The curves illustrate the response of each of the two microphones. Comparison of the two curves shows the frequency response of the Electret microphone to be within 2 dB compared to the B&K microphone. The Electret exhibits some roll off below 30 Hz which is due to the highpass filters which are used in conjunction with the lowpass anti-aliasing filters discussed earlier. The Electret microphones shows less agreement above 270 Hz which is acceptable because the intended operational range of the system and the spectral content of the sound as discussed in Section 3.2. The resulting spectrum does not look like broadband white noise due to the electro-acoustic response of the speakers and microphones, and the acoustic response of the reverberant enclosure in which the testing is performed. The Electret microphones are not calibrated for two reasons. The goal of the system is to reduce the signal from the feedback microphones to a minimal level and the absolute level does not effect the results. The other reason calibration is not performed is because different gain settings are used as needed when data is collected and used during testing.

The feedback microphones are attached to the cabin interior in the vicinity of the co-pilot's head as is described in the 'Data Collection' and 'Acoustic Testing' sections. The feedforward microphone is placed inside a lead enclosure as illustrated in Figure 3.3.2. The lead enclosure consists of a lead cube (approximately 1.25 inches per side) with a hole drilled approximately halfway through into which the microphone is placed. The lead enclosure is surrounded by a block of closed cell polyurethane foam to further isolate the microphone from vibrations and cabin noise. The microphone is also placed in a foam liner inside the lead block to ensure it does not contact the enclosure. The feedforward microphone is placed in close proximity to the front windshield as shown in the figure and is held in position by duct tape. The system is always removed and stowed for takeoff and landing.

These microphones are used to collect data during flight in a Piper Malibu Mirage, Piper Cherokee and Cessna 172 Skyhawk aircraft. The same microphones are used for all data collection and testing.

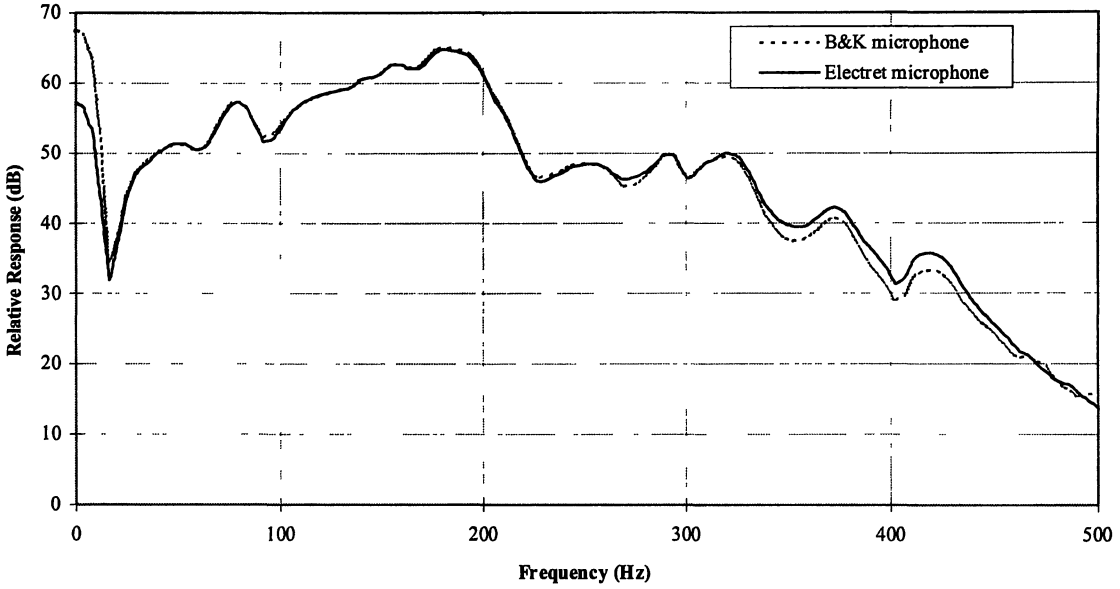
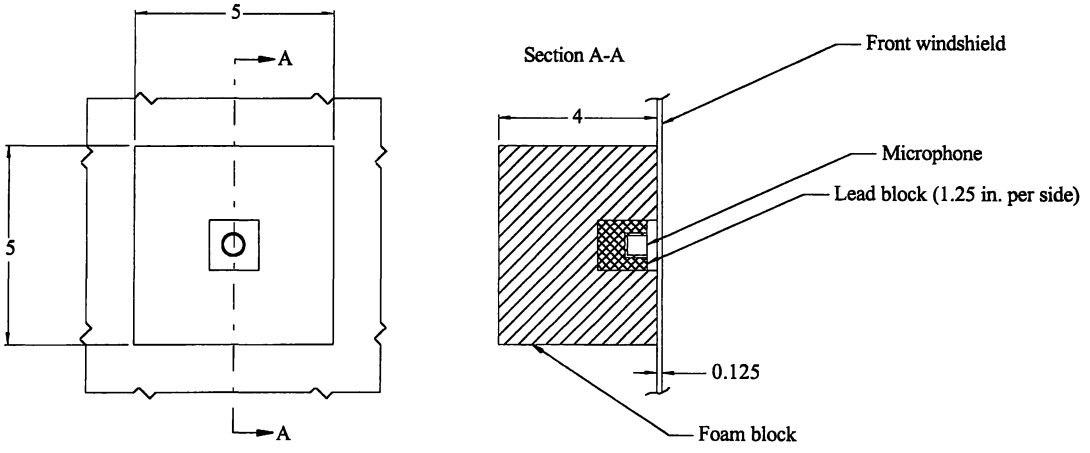


Figure 3.3.1 Comparative test results for the Electret microphone and the B&K microphone



Note: 1. Not to scale
2. All dimensions in inches (in.)

Figure 3.3.2 Schematic to illustrate the feedforward system and implementation

3.4 AMPLIFIERS AND SPEAKERS

Implementing an ANC system in an aircraft requires a careful analysis of the cabin response, cabin noise levels and spectra, power requirements and practical considerations such as the location and allowable size of the speaker enclosures. Due to the scope of this research, readily available speakers and amplifiers are used. The secondary source speakers and amplifiers are micro-component stereo items. The appendix details the make and model of each item. The speaker enclosures are approximately 6 in. by 7 in. by 10 in. and the speaker/enclosure combination are verified to have reasonable frequency response characteristics. This speaker/enclosure combination is selected in part due to the small size which allows a variety of speaker placements during testing. For the purposes of this research these components are satisfactory.

CHAPTER 4
TESTING AND ANALYSIS

This chapter presents the cabin noise and reference signal data recorded during flight in three different aircraft. Electronic system simulation results are presented and analyzed. The single input, single output (SISO) system testing methods are outlined and the results are analyzed. The results obtained with the SISO system tests influence the multiple input, multiple output (MIMO) systems tests as described in the last section of this chapter.

4.1 ELECTRONIC SIMULATION

A simple electronic simulation is used to assure the algorithms are correctly implemented and interfacing correctly the A/D and D/A boards. The system consists of resistors which are used to allow electronic addition of the primary and secondary signals. The resistors simulate the primary and secondary path transfer functions by replacing the amplifiers, speakers and microphones in Figure 2.9.1 in Section 2.9.

The system identification is tested through the use of this circuit and the resulting FIR weights are analyzed after identification is complete. Recall that the secondary path consists of the electro/acoustic transfer function between the output of the D/A board and the input to the A/D board. In this simple simulation the secondary path consists only of the input and output filters so the identified system response should correspond to these transfer functions. Mathcad is used to process the resulting FIR weights to determine the frequency response modeled by the system identification.

A function generator is used to apply a signal to the circuit input to serve as the noise and reference signals. Simple sine waves, saw tooth waves, pulse trains and square waves are all used to test the system. During testing an oscilloscope and a spectrum analyzer are used to monitor the system performance. The results of these tests are presented below.

4.2.1 Electronic Simulation Results

The only secondary path effects present in this system are due to the filters and the boards which allow the system identification results to be compared to the known filter transfer functions to evaluate the performance of the identification system. Figure 4.1.1 shows the resulting identification FIR weight values as a function of the weight number (i.e. the impulse response) after the MIMO identification is complete. Thirty weights are used for each secondary path FIR model. The MIMO system has two secondary sources and two feedback sensors which means there are four secondary paths and four curves in the figure. The impulse response exhibits a delay of five samples before producing any output and characteristics associated with a lowpass filter. Because the impulse response is not easily interpreted and the frequency domain transfer function provides additional information, the secondary path FIR filter model weights are used to evaluate the frequency domain transfer functions through the use of Mathcad.

The resulting transfer function gain is illustrated in Figure 4.1.2. Once again there is one curve corresponding to each of the four secondary paths in the figure. The differences between the curves are due to slight differences in the electronic circuitry for each path but the general trends are the same. The transfer function gain has the characteristics of a high order, lowpass filter with a corner frequency at approximately 350 Hz with some gain in the passband. The calculated transfer function corresponds to the filter settings used during the system identification which indicates the system identification is working as designed. The SISO electronic simulations achieve similar results.

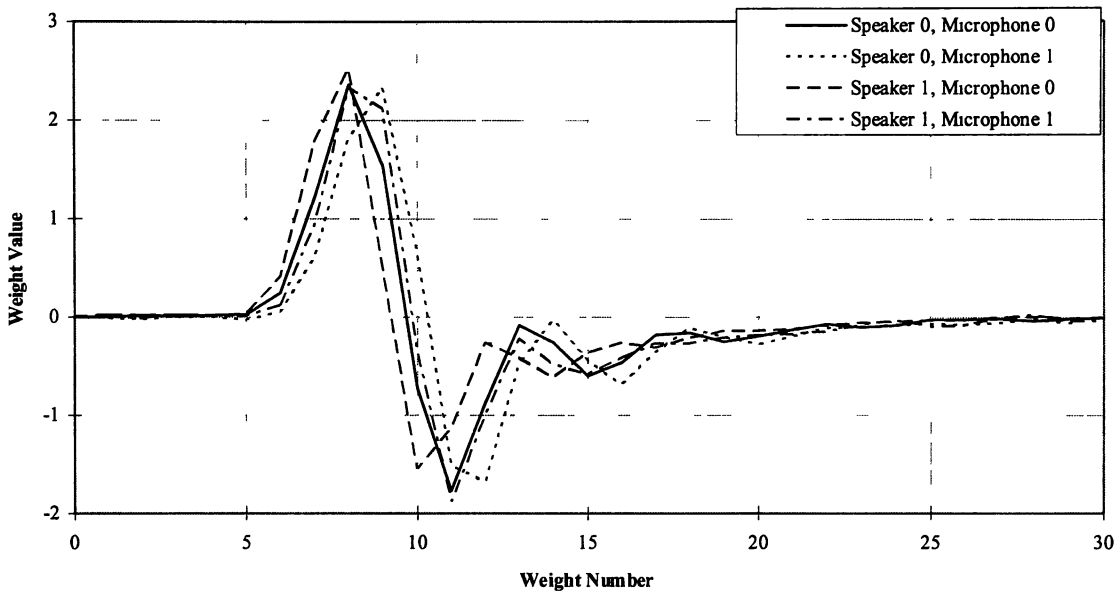


Figure 4.1.1 System identification impulse response obtained during an electronic system simulation

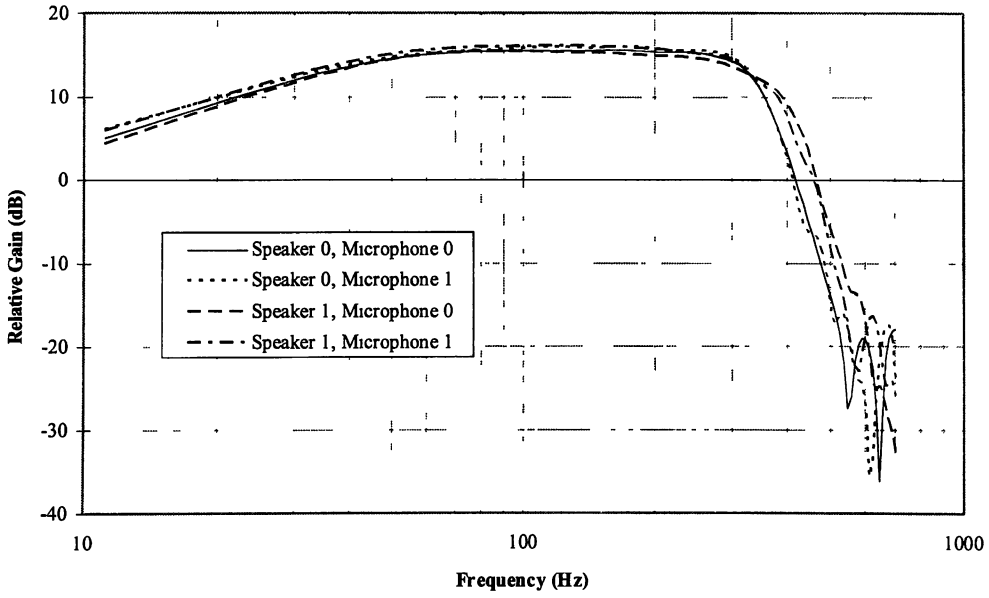


Figure 4.1.2 System identification secondary path response transfer function gain

Figure 4.1.3 illustrates the feedback sensor spectra before control and during control when a filtered ramp function is applied to the input of the MIMO system to simulate the noise. A Hanning window is used in the spectral analysis. The RMS average of 8 consecutive ensembles are used resulting in a spectral bandwidth of 3 Hz. The control system FIR filter consists of 30 weights per secondary source. The figure shows the four most dominant tones reduced by significant amounts but the system does not reduce any of the other tones and increases the amplitude of these tones. Control beyond 400 Hz can not realistically be expected since the system is designed to implement control below 400 Hz. The use of 30 weights resulted in very rapid convergence (less than half a second) and the large reductions indicated. Further reductions are not meaningful since this system is only a simulation. If similar reductions are achieved acoustically, additional reductions below those indicated will be virtually unnoticeable unless the other tones are also reduced. Similar performance is achieved with the single seat system simulation. An investigation of the controller transfer function helps to explain these results. The controller impulse response is used to evaluate the frequency domain transfer function by the same method as used to evaluate the identification transfer function.

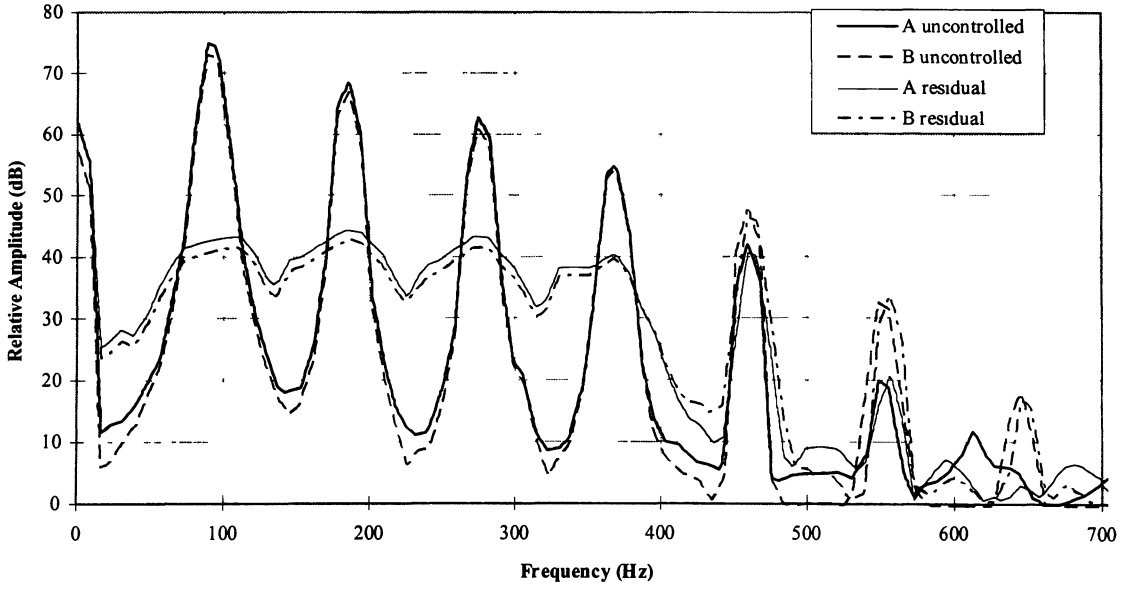


Figure 4.1.3 Spectra at feedback sensors 0 and 1 before controlling a filtered ramp function and the residual noise during control

The frequency domain transfer function gains are calculated and the results are plotted in Figure 4.1.4. There are two transfer functions in the figure corresponding to one for each of the two controllers. The controller transfer function is significantly more complex than the identification transfer function but notice the largest gains are at the same frequencies as the tones. The identification results in Figure 4.1.2 show the filters roll off between 350 Hz and 370 Hz which means the control algorithm must implement increasingly larger gains to control tones beyond the corner frequency. This is seen by examining the transfer function gain and noticing the large gain necessary to reduce the tone closest to 400 Hz. Higher frequency tones are not controllable because the algorithm is unable to construct a filter with the necessary transfer function characteristics at the higher frequencies without sacrificing control of the lower frequencies increasing the error.

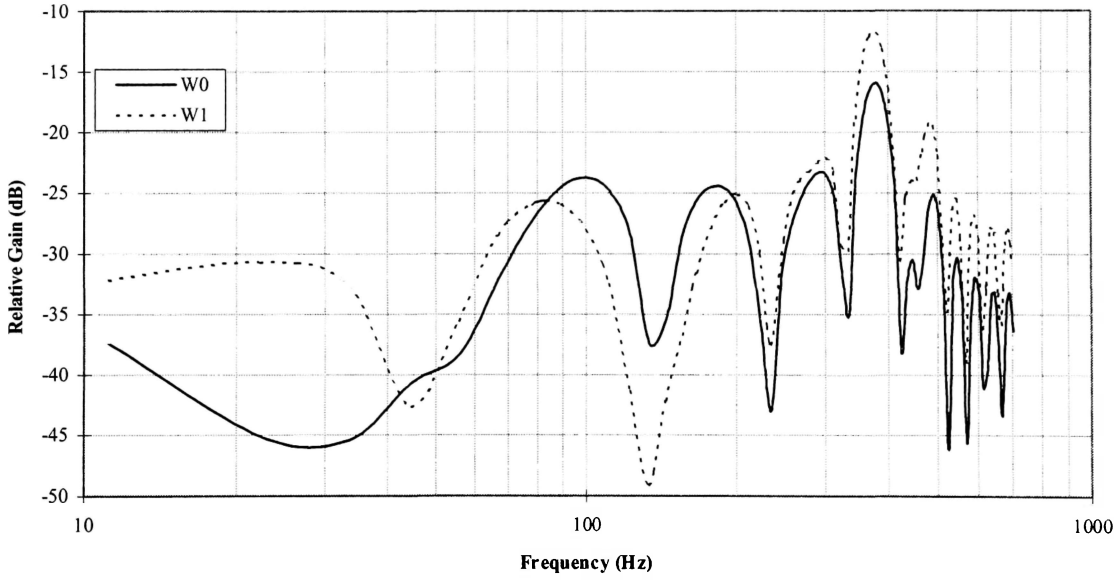


Figure 4.1.4 Controller transfer function gain

Both the identification and control systems are shown to function as designed. Both systems perform as expected in the frequency range for which they are designed. The control system performance beyond the designed frequency range is also as expected. With the knowledge that the systems are correctly implemented, acoustic testing is possible to evaluate the ability of the system to reduce cabin noise.

4.2 DATA COLLECTION

Data from several different aircraft are used throughout this research during both the design and testing of the system. Piper Saratoga cabin noise spectra provided by Piper Aircraft and Cessna 172 spectra (Ref 7) are the primary sources used during the design phase. Piper Malibu, Piper Cherokee 140 and Cessna 172 flight data recordings are used during laboratory simulations to test the system performance.

The feedforward microphone package is affixed to the front windshield with duct tape in a position directly in front of the copilot's position. During some flights, data is collected with the feedforward microphone in different positions on the front windshield. During Cherokee flights, the feedback microphone is affixed to the cabin headliner with the microphone positioned above the head of the co-pilot position. In some instances (Malibu and Cessna) the feedback microphone is attached to the seat, close to the inside shoulder of one of the passengers or is clipped to the passenger's collar. The reference signal and feedback microphone signal are recorded simultaneously through the use of an analog data recorder. The appendix lists the recording devices used during these flights.

Straight and level flight with nominal cruise throttle positions are used to collect all data. Climb data is also collected in the Piper Cherokee 140 and Malibu Mirage. All data is collected during smooth atmospheric conditions with minimal turbulence except the Malibu data which is taken during gusting conditions with some turbulence. The recorded data is analyzed in the laboratory through the use of the spectrum analyzer. All data collection is for the SISO system in that only one feedback microphone is used to provide cabin noise data.

4.3 DATA ANALYSIS

This section presents the data collected as described in the previous section. For each of the aircraft the reference signal spectrum and the cabin noise spectrum are displayed on the same plot. The spectra in the figures are obtained through the use of a spectrum analyzer. A Hanning window is used in all the spectral analysis. The RMS average of 8 consecutive ensembles are used with a spectrum bandwidth of 6 Hz. During the spectral analysis it is noted that the amplitudes of the harmonics are not constant with time and vary by several decibels over a time.

The spectra of the reference signal and the cabin noise recorded during flight in a Cessna Skyhawk 172 are illustrated in Figure 4.3.1. From the figure, notice the dominant BPF tone at approximately 83 Hz and the BPF harmonics at 166 Hz, 250 Hz and 333 Hz. The amplitude of the first harmonic of BPF is roughly 10 dB below the BPF tone in both spectra. The second BPF harmonic (250 Hz) is below the first harmonic in the cabin noise but is dominant over the first harmonic in the reference signal indicating reduced coherence at this frequency. Also notice the difference in the amplitude of the third harmonic in both signals.

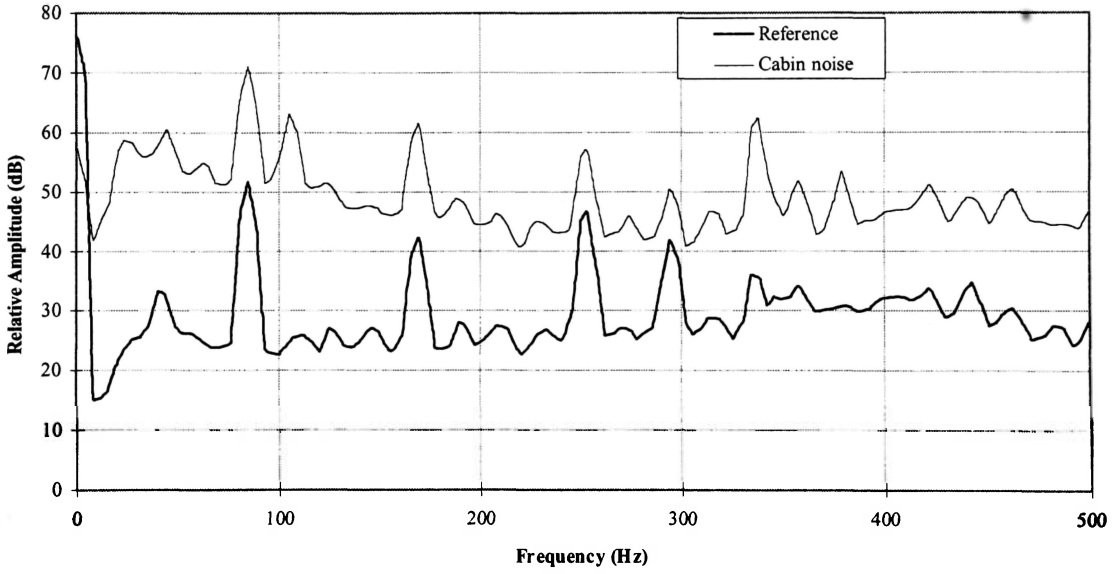


Figure 4.3.1 Cessna Skyhawk 172 cabin noise and reference signal spectra during cruise

There is a tone at approximately 110 Hz in the cabin noise which is absent from the reference signal in the figure. This tone is not present in the cabin noise at all times during cruise and the cause of this tone is not known. Earlier research performed in the same aircraft (Ref. 7) reported a similar tone but did not conclusively determine the source of the tone. The reason this tone is intermittently present in the data is not determined. The tone is one of the most dominant frequencies and because of the poor coherence at this frequency, minimal control is expected. The reference signal is also poorly coherent with some of the significant tones below 50 Hz.

The cabin noise and reference signal spectra for a Piper Cherokee 140 during cruise and climb are illustrated in Figures 4.3.2 and 4.3.3 respectively. Considering the cruise spectra first; BPF and the lowest two harmonics of BPF are clearly the dominant tones in the cabin noise. Some lower frequency cabin noise below 50 Hz is also significant. The BPF component in the reference signal correlates very poorly with the cabin noise component during cruise. Notice the climb spectra shows better coherence between the signals at BPF and the first harmonic of BPF. During climb the coherence at the second and third harmonics is still weak.

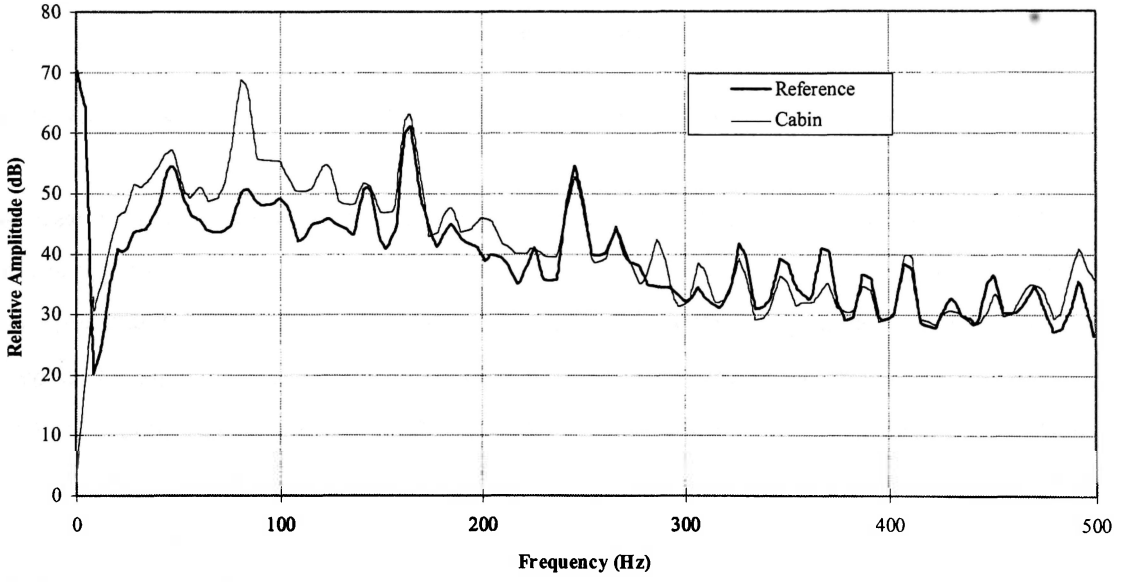


Figure 4.3.2 Piper Cherokee 140 cabin noise and reference signal spectra during cruise

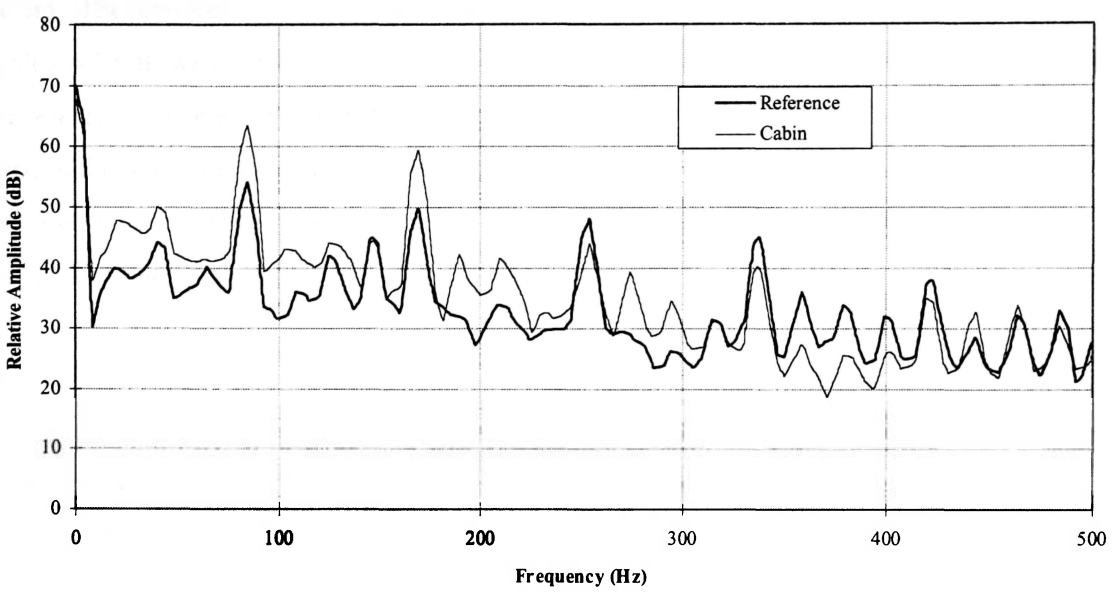


Figure 4.3.3 Piper Cherokee 140 cabin noise and reference signal spectra during climb

The reason the reference signal shows poor coherence at BPF with the cabin noise is not certain but it is hypothesized that the Cherokee fuselage structural characteristics are responsible. The front windshield on the Cessna 172 is one continuous piece without any supporting structure except at the edges. The Cherokee 140 front windshield is constructed in two pieces with a structural support joining the two windshield halves in the middle. The Cherokee 140 windshield also has more curvature than the Cessna 172. The net result of the structural support member and the curvature is to make the Cherokee 140 front windshield(s) more rigid at the lower frequencies compared to the Cessna 172 windshield. The increased rigidity reduces the windshield deflections at lower frequencies such as at BPF which partly explains the low BPF amplitude in the reference signal. During climb, the propeller experiences greater loading which causes stronger pressure fluctuations in the wake of the propeller, particularly at BPF. The stronger pressure fluctuations cause larger windshield deflections at BPF and the BPF tone is clearly dominant in the climb reference signal spectrum in Figure 4.3.3.

Figure 4.3.4 shows the Piper Malibu Mirage cabin noise and reference signal spectra during cruise conditions. The cabin noise is dominated by the BPF tone at approximately 85 Hz and an engine tone at approximately 125 Hz. Also note from the figure the general shape of the cabin noise

spectra. The reference signal spectrum shows the first harmonic of BPF to dominate the BPF tone by almost 15 dB which indicates very low coherence with the cabin noise at both these frequencies. The reference signal tones cohere poorly with the cabin noise at all frequencies and the general shape of the reference spectrum is also different from the cruise spectrum. For example the BPF tone is dominated by the tones between 250 Hz to 300 Hz in the reference signal which is not the case in the cabin noise.

The cabin noise and reference signal spectra during climb in the Malibu Mirage (Figure 4.3.5) show additional dominant tones in the cabin noise. Investigation of the figure shows the amplitude of the tones in the cabin noise and reference signals are significantly different indicating low coherence between the two spectra.

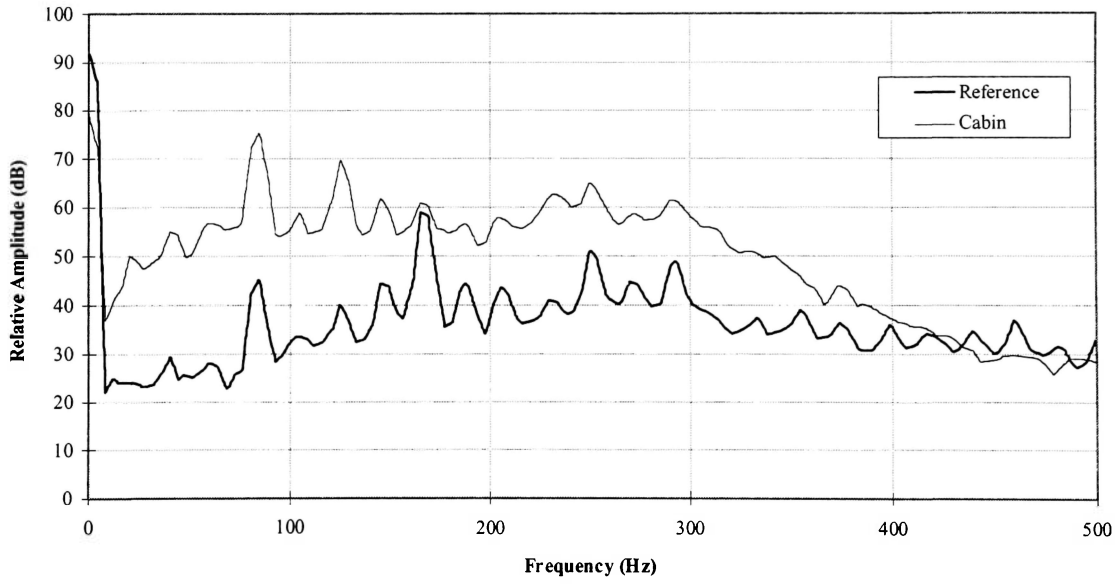


Figure 4.3.4 Piper Malibu Mirage cabin noise and reference signal spectra during cruise

The Malibu Mirage front windshield structure is similar to the Cherokee structure in that the front windshield consists of two halves with a longitudinal structural member in the middle. Additionally, the Malibu front windshield is made of 1/4 inch thick Plexiglas (Ref. 2) compared to the 1/8 inch used in the Cherokee and Cessna. In the Malibu Mirage the pilot's front windshield is made of glass (Ref. 2) which also reduces vibrations in this windshield. The Malibu Mirage is

pressurized which requires a stronger fuselage structure to withstand the pressurization loads. Pressure loads are also a consideration for all door and window seals and retaining structures. The resulting fuselage structure and front windshields are more rigid, reducing the role of the windshields as sound transmission paths.

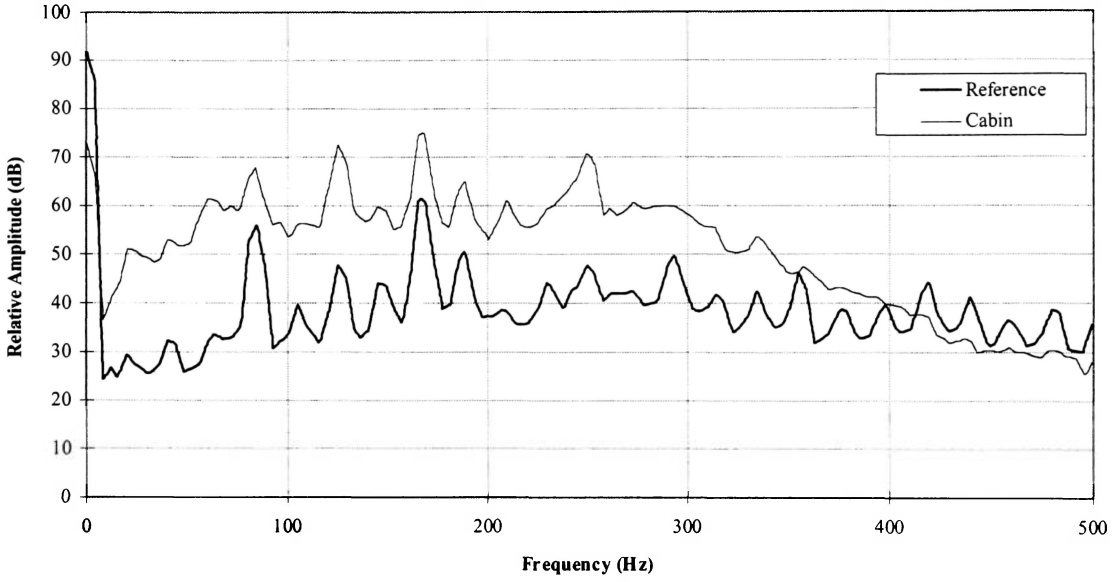


Figure 4.3.5 Piper Malibu Mirage cabin noise and reference signal during climb

4.4 ACOUSTIC TESTING

The SISO system is acoustically tested to evaluate the system performance with the reference signal and cabin noise recorded during flight. The SISO test results are presented and analyzed before the MIMO system testing is described. Reference signal effects are discussed and based on the SISO results, the MIMO system test methodology and setup are detailed.

4.4.1 SISO Acoustic Testing

With the equipment and facilities available it is not possible to evaluate the system performance during flight except in the Malibu Mirage. The main reasons are the combined bulk of the equipment and the need for 120 V, 60 Hz power for the amplifiers, oscilloscope, spectrum analyzer and the laptop docking station. The system is tested in laboratory simulations through the use of

the feedforward and cabin noise data recorded during flight as discussed earlier. Ideally, in a static test the system is installed in the aircraft cabin and speakers are used to simulate the cabin noise. However, due to the availability of the aircraft, such a simulation is not possible.

Initial testing in a reverberant laboratory reveals the acoustic response of laboratory reduces the coherence between the reference and feedback signals by an excessive amount. Testing in an anechoic chamber will provided idealized results due to the dissipative nature of enclosure. Presently there is no anechoic chamber available at the university. An aircraft cabin is a fairly diffuse, lightly damped enclosure due to the shape and structure of the cabin. In an effort to test the system in a diffuse, lightly damped enclosure, all acoustic testing is performed inside an automobile cabin.

The purpose of these tests is not to determine the speaker and microphone placement needed to obtain global reductions in each of the different aircraft or in the testing enclosure. Rather, these tests are used to evaluate the ability to acoustically reduce cabin noise with a given reference signal and to obtain some experience with the acoustic coupling between the sources and microphones in the case of the MIMO system. Two different reference signal scenarios are tested. An analysis of the data collected during flights in the different aircraft show coherence problems between the reference and feedback signals in all aircraft. It is necessary to test how well the system can reduce the cabin noise with the associated reference signal to determine if the feedforward method is feasible. The second method consists of using the cabin noise recording as both the reference signal and the signal sent to the primary speakers to simulate a reference signal with strong coherence.

The cabin noise data is passed through an audio amplifier and is used to drive an audio speaker, placed in the car, exciting the primary noise in the cabin. This primary speaker is placed between the front seats and is aimed at the front windshield. The secondary speaker is placed on the cabin floor behind the seat backrest of the front passenger and is aimed up towards the ceiling. The feedback microphone is attached to the cabin ceiling above the front passenger position and is secured with duct tape. The sound pressure level meter is placed on top the front passenger headrest with the microphone unit near the feedback microphone. The sound pressure level meter

is used to measure the OASPL during the tests. The speakers and microphones are the only components placed inside the car. All personnel and equipment are outside the vehicle.

The reference signal and cabin noise collected in flight are used as follows: the reference signal is passed through the feedforward filter and the resulting signal is used as the reference signal input to the A/D board. The cabin noise signal is passed to the primary speaker after amplification. The appendix lists the amplifiers and speakers used during testing. The secondary source and feedback microphone gains are adjusted as needed.

After system identification is complete, the recorded data is used to generate cabin noise through the use of the primary speaker and the control system is activated. 60 weights are used for both the identification and control filters. The feedback microphone output is also attached to an oscilloscope to monitor the performance in the time domain and a spectrum analyzer to observe the performance in the frequency domain. The spectrum analyzer is used to store spectra before and during control. A Hanning window is used with a 3 Hz bandwidth. The spectra are stored after taking the RMS average of 8 ensembles. Testing is performed with Cherokee, Cessna 172 and Malibu Mirage flight data.

It is desirable to test the system performance when a reference signal is used which coheres closely with the cabin noise. To test the system when there is strong coherence between the reference signal and the cabin noise, the cabin noise recorded during flight is used as both the reference signal and the cabin noise. The same test setup and data collection method is used. These tests are performed for the Piper Cherokee and the Cessna 172. Due to time constraints on the equipment loan which enabled this research and equipment constraints, the Malibu data is not tested.

The single seat system testing results are presented below because the results effected the MIMO system testing. The high coherence simulation testing results are presented in the Chapter 5.

4.4.2 SISO Acoustic Testing Results And Analysis

Figure 4.4.1 illustrates the Cessna 172 cabin noise spectrum before control and the residual noise spectrum when control is implemented. Approximately 8-9 dB OASPL reduction is achieved in this test. As illustrated in the figure, the BPF tone is reduced by over 20 dB and the lowest two harmonic tones of BPF are reduced by 9 dB and 8 dB respectively. Notice the stray tone at approximately 110 Hz is not reduced. Comparison of the residual noise spectrum with the reference signal spectrum in Figure 4.3.1 shows the stray tone is absent from the reference signal which is why the tone is not controlled. The low frequency noise below 60 Hz is not reduced due to the lack of coherence between the reference signal and the cabin noise.

Attempts to control the Cherokee and Malibu Mirage cabin noise both fail due to the lack of coherence between the reference signal and the cabin noise as discussed earlier in Section 4.3.

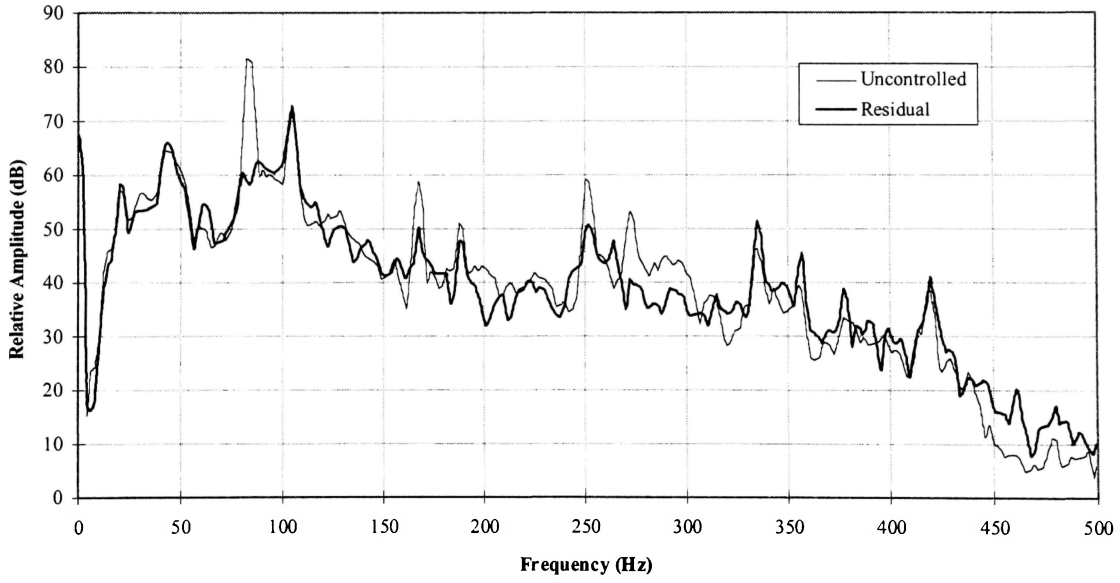


Figure 4.4.1 Cessna 172 cabin noise spectra before control and residual noise during control

The current feedforward system only detects sounds which are excited by the front windshield of the cabin. The dominant driver for the windshield vibrations is a combination of the acoustic pressure disturbances caused by the propeller and structural borne vibrations. The windshield is a poor transmission path for structural borne vibrations including those due to engine vibrations or

other structural borne vibrations. Any dominant tones in the cabin which are due to engine vibrations are absent from the reference signal and these tones can not be reduced. The inability to reduce these engine tones is a serious drawback since both the Cessna 172 and the Malibu Mirage have an engine tone which is almost as dominant as the BPF tone (Section 4.3).

The other major problem with this feedforward method is that in some aircraft the windshield and supporting structure is sufficiently rigid that the lower frequency vibrations are not passed as readily as the higher frequency vibrations and the resulting reference signal is weakly coherent with the cabin noise. This is the case for both the Cherokee 140 and the Malibu Mirage, and control is not possible with the resulting reference signal.

A potential problem with the current feedforward method is that the sound from the secondary sources is detected by the feedforward microphone which corrupts the reference signal. Testing shows this corruption to be relatively small and has minimal effect on the system performance for this research. However, such a system is susceptible to corruption in some applications may be unsuitable for implementation in aircraft.

The math modeling shows a strong coherence is needed to enable significant reductions. The control system modifies the entire spectrum of the reference signal to generate the control signal and the system is only able to apply a limited amount of gain at one frequency before losing control at other frequencies, increasing the error. Since the algorithm functions to reduce the error in the overall mean square sense, weakly coherent tones can not be controlled because doing so will not reduce the mean square error.

Part of the reason the acoustic feedforward method is used is to determine if any broadband reduction is possible and whether the resulting broadband reduction is large enough to be noticeable in the OASPL reduction. As seen in Figure 4.4.1, there is no broadband reduction. Since broadband reductions are not achieved under these conditions and there are significant drawbacks to the current feedforward method, different feedforward methods are investigated as described in the "Feedforward Considerations" and "Math Modeling" sections.

After additional math modeling it is decided to test the two seat system with a tonal reference signal. A lack of lowpass filters to perform the necessary filtering to allow MIMO testing to simulate a high coherence reference signal (as in the SISO tests) is an additional reason that only tonal control is tested. Collecting such a tonal reference signal in flight requires the design, building, testing and flight test certification of a new feedforward system. Due to the limited scope of this research, tonal reference data and the associated cabin noise data can not be collected during flight. Testing the system with a tonal reference signal will allow evaluation of a tonal reference signal method for future work. The MIMO system testing is described in the following section

4.4.3 MIMO Acoustic Testing

An experimental setup similar to the SISO setup is used. All acoustic testing is performed inside the cabin. The feedback microphones are taped above two of the seats. The primary speaker is placed in the same position as in the SISO tests. The secondary speakers are placed in several different positions to evaluate the performance when the acoustic modes are excited from different locations.

As discussed under "Reference Signal Considerations" in Section 2.8, two types of tonal feedforward are possible. The most versatile method involves several reference signals with each signal limited to contain one tone. As many reference signals are used as there are tones to control. The second method consists of using one reference signal which contains all the necessary tonal components. To test the multiple reference signal method requires several function generators to produce the tones and extensive modifications to the identification and control algorithms necessary to use the multiple reference signals. Due to time and equipment constraints it is not possible to implement the necessary modifications to test the performance of a multiple reference signal system. For this reason a single tonal reference signal is tested.

A ramp function signal obtained from a function generator (listed in Appendix B) is filtered to generate several tones which decrease in amplitude with increasing frequency. The same signal is used as both the reference signal and the cabin noise signal sent to the primary speaker. This method is used to simulate cabin noise consisting of a BPF component and several harmonics. The

reference signal therefore exhibits strong coherence with the cabin noise tones and simulates a tonal reference signal. Several different secondary speaker locations are tested as a quick method to determine the best speaker locations to obtain equal reductions at both speakers. The results of these tests are presented in the Chapter 5.

CHAPTER 5
RESULTS

The single input, single output (SISO) results are discussed in the previous chapter and are summarized in this chapter. The results for the SISO testing to simulate a reference signal which has strong coherence with the cabin noise is presented in this chapter and the multiple input, multiple output (MIMO) testing results are also discussed.

5.1 SISO SYSTEM TESTING RESULTS

As mentioned in the previous chapter, approximately 8 dB - 9 dB overall sound pressure level (OASPL) reduction is achieved with the cabin noise and reference signal recorded in a Cessna 172 during cruising flight. Figure 5.1.1 illustrates the noise spectrum before control and during control. Notice the different frequency scale and the slightly different spectral content compared to Figure 4.4.1. The different spectral content in these two figures is because the tones in the spectrum vary slightly over time and the data used to test the system to obtain Figure 5.1.1 is at a different time from that used for Figure 4.4.1. The figure shows that the engine tone at approximately 110 Hz is not reduced by any significant amount due to the virtual absence of this tone from the reference data (Figure 4.3.1). Notice there is no broadband reduction which is discussed in some detail later.

The results obtained with the Cessna flight data, and the inability to reduce the Piper Cherokee 140 and the Piper Malibu Mirage cabin noise, illustrate the problems with the acoustic feedforward method. Engine tones and other structural borne vibration tones are unlikely to appear in the reference signal. There is insufficient coherence between the broadband components of the reference signal and the cabin noise to enable a reduction in the broadband component of the cabin noise.

Due to the problems with the current feedforward method, the MIMO system testing is altered to test a tonal reference signal scenario as described in the previous chapter. Before the MIMO results are discussed, the SISO results obtained during testing to simulate a high coherence reference signal are presented.

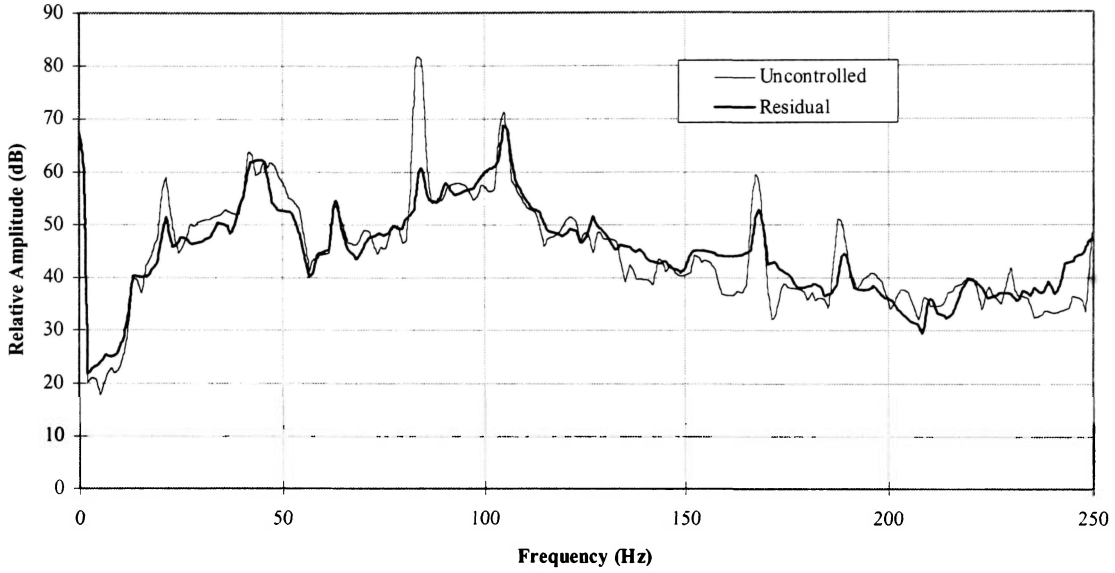


Figure 5.1.1 Spectra of Cessna 172 cabin noise before control and residual noise during control

5.2 SISO HIGH COHERENCE TESTING RESULTS

When a reference signal is used which has close coherence with cabin noise, the Cherokee cabin noise is successfully reduced by approximately 12-15 dB OASPL. There is some difference between the reference signal and the resulting cabin noise due to the speaker electric/acoustic response and the cabin acoustic response but the coherence is still strong at the dominant tones. The primary source of these reductions is the roughly 20 dB reduction at BPF and greater than 15 dB reduction at the first harmonic as shown in Figure 5.2.1.

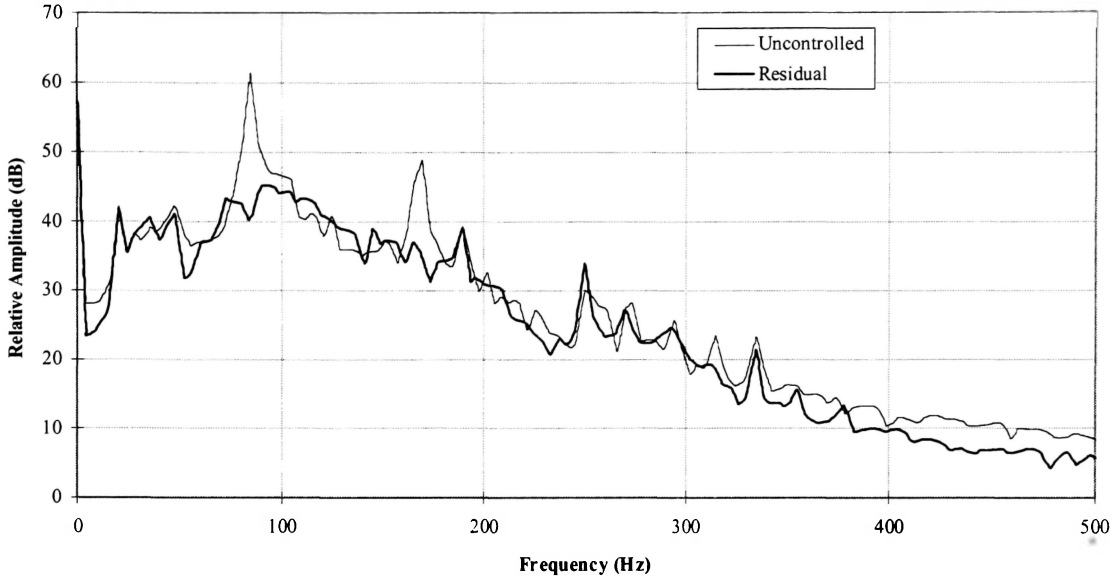


Figure 5.2.1 Piper Cherokee cabin noise spectrum before control and residual noise during control with high coherence reference signal.

The Cessna cabin noise is also reduced by approximately 12-15 dB OASPL. Notice the engine tone at approximately 110 Hz is now reduced by approximately 12 dB (Figure 5.2.2). A comparison of Figure 5.2.2 with Figure 4.4.1 shows the reductions at BPF and the first harmonic are relatively unchanged while a large reduction of the engine tone is now achieved. There are some differences between the two spectra at the higher harmonics because the tests are performed with data which is recorded at different times during cruising flight. Malibu Mirage data is not presented

It is interesting to note that there is some broadband reduction above 350 Hz to 400 Hz in Figures 5.2.1 and 5.2.2. This has an insignificant affect on the OASPL reduction because the original amplitudes are far below the lower frequency broadband amplitudes which are not reduced. The results are interesting because the system design is expected to obtain lower frequency broadband reductions before any higher frequency broadband reductions are obtained. Due to time and equipment constraints the system is not tested to evaluate the ability of the system to reduce purely broadband noise.

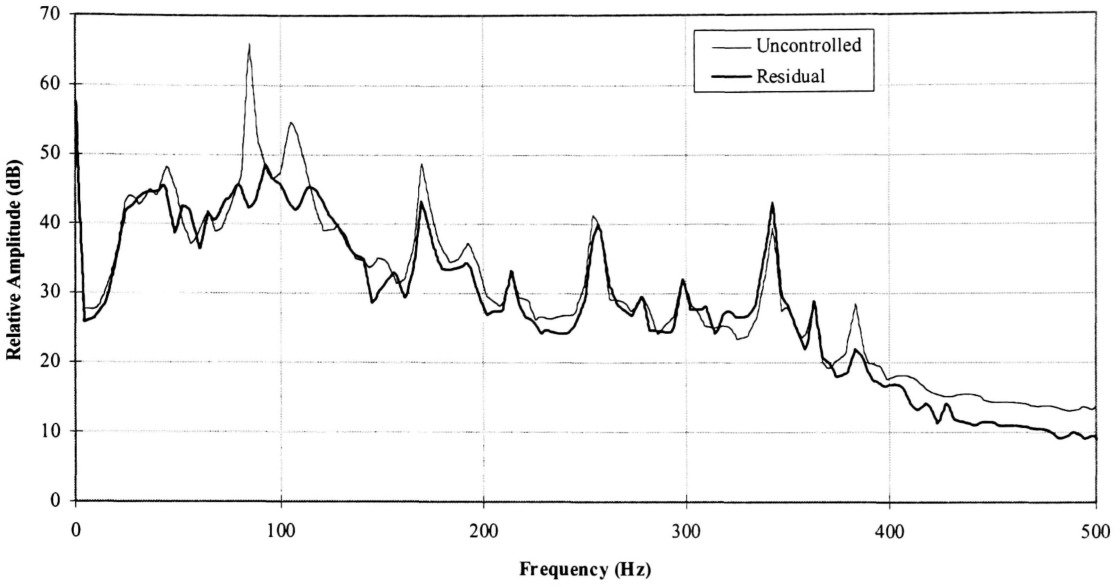


Figure 5.2.2 Cessna 172 cabin noise spectrum before control and residual noise during control with high coherence reference signal.

There are three possible explanations as to why there is no lower frequency broadband reductions. The system must be causal to control random noise which means that the delay associated with the control signal path must be less than the transport time between the reference signal and the feedback microphone. The feedforward sensor must pass the reference signal to the control system which outputs the corresponding control signal which must travel from the secondary sources to the feedback microphone before the noise travels from the feedforward sensor to the feedback microphone. The control signal delay is composed of the transport delay (between secondary sources and feedback microphones) and the calculation (group) delay. The transport delays are verified to be causal before data collection and system tests. However, the group delays may have exceeded the transport delays which causes the system to be non-causal. If the system is not causal, broadband reductions are only possible if the broadband is tonal.

The second possible reason is that the filters are insufficient and the resulting aliasing corrupts the lower frequencies by an excessive amount. While there is certainly some aliasing, this is probably not the cause for the lack of lower frequency broadband reduction since the aliased data is expected to have minimal impact as is discussed in the filter section.

The third possible reason for the lack of broadband reductions is explained by considering the controller transfer function. Figure 5.2.3 illustrates the transfer function gain associated with the SISO identification performed before applying control in these tests. The transfer function is not as flat as desired with rolloff on either side of the peak approximately 100 Hz. This secondary path response is due to the electric/acoustic response of the system which is affected in large part by the acoustic response of the enclosure. The acoustic response of the enclosure also alters the cabin noise introduced to the cabin by the primary speaker as detected by the feedback microphones. The result is reduced coherence between the reference signal and the feedback signal which prevents the control of the broadband noise. The reason the higher frequency broadband reductions are achieved is not certain considering the secondary path transfer function gain is approximately 30 dB to 50 dB below the gain at the 100 Hz peak in Figure 5.2.3. The higher frequency broadband reductions may be caused by the acoustic response of the enclosure, and the placement of the secondary speaker and microphone.

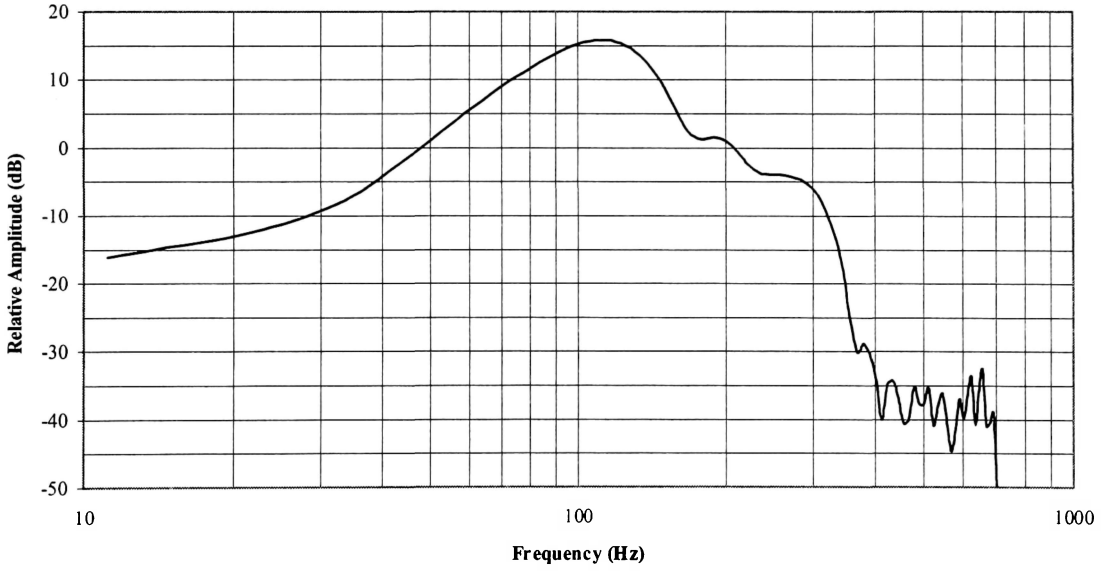


Figure 5.2.3 SISO secondary path response transfer function gain

These tests illustrate the difficulty of obtaining a reference signal which has sufficient coherence with the cabin noise to obtain broadband reductions in this type of aircraft. The tests also show the difficulty in simulating cabin reductions in the laboratory, even if a fuselage is available for testing.

If a fuselage is available for testing, there is still the difficulty associated with reproducing the cabin noise accurately such that the same coherence exists between the reference signal and the cabin noise during the tests as exists during flight.

The above results are obtained before any MIMO testing. The results show the current feedforward method to have too many problems and obtaining a good reference signal with the required coherence is difficult in this type of aircraft. Obtaining all the propeller and engine noise in the right proportions in one feedforward signal is difficult for this type of aircraft and even if a solution is found for one aircraft, the method will not necessarily work for all aircraft of this type. The use of two broadband reference signals, one dominated by the propeller and the other dominated by the engine is considered but obtaining the necessary coherence through the use of this method is doubtful. Since the primary objective is to reduce the dominant tones, a tonal reference signal is an attractive alternative to the current feedforward method. As mentioned previously, multiple tonal reference signals offer the most flexibility but such a method requires an equal number of additional A/D channels and extensive modification to the control and identification programs. Due to time and equipment constraints, a single tonal reference signal is tested to evaluate system performance with such a reference signal. The results are presented in the following section.

5.3 MIMO TESTING RESULTS

The problems associated with the current feedforward method as evaluated through the use of the SISO system prompted the revised testing method for the MIMO system. The MIMO testing attempts to simulate a tonal reference signal which exhibits good coherence with the cabin noise.

The MIMO acoustic tests include testing the system with the secondary speakers at several different locations. These tests show the performance to be very sensitive to the speaker placement. Most locations only allow the system to reduce the lowest frequency, largest amplitude tone and several positions are tested before a suitable location is obtained which results in better reductions. This sensitivity to speaker location is due to the acoustic response of the enclosure and the acoustic coupling between speakers and microphones. Figure 5.3.1a shows the reference

signal and the cabin noise spectrum at each of the two feedback microphones. The graphs in Figure 5.3.1b and Figure 5.3.1c show the cabin noise spectrum before control and during control at feedback microphone number 1 and 2 respectively. Figures 5.3.1b and 5.3.1c show the primary tone is reduced by approximately 20 dB at both microphones and the first harmonic is reduced by 9 dB at both microphones. The second harmonic is reduced by approximately 2 dB at both microphones. The resultant OASPL reduction is measured at 17 dB to 18 dB.

Notice from Figure 5.3.1a the difference between the reference signal and the cabin noise at each of the microphones, particularly below 80 Hz and above 250 Hz. This difference is due to the electric/acoustic response of the primary speaker and the acoustic response of the enclosure. Larger reductions can be achieved if the reference signal is made to cohere more closely with the cabin noise but this is not achieved in these tests. The reductions achieved with the tonal reference signal show the system to work effectively with such a signal and shows future work with a tonal reference signal to be an effective alternative to the current feedforward method.

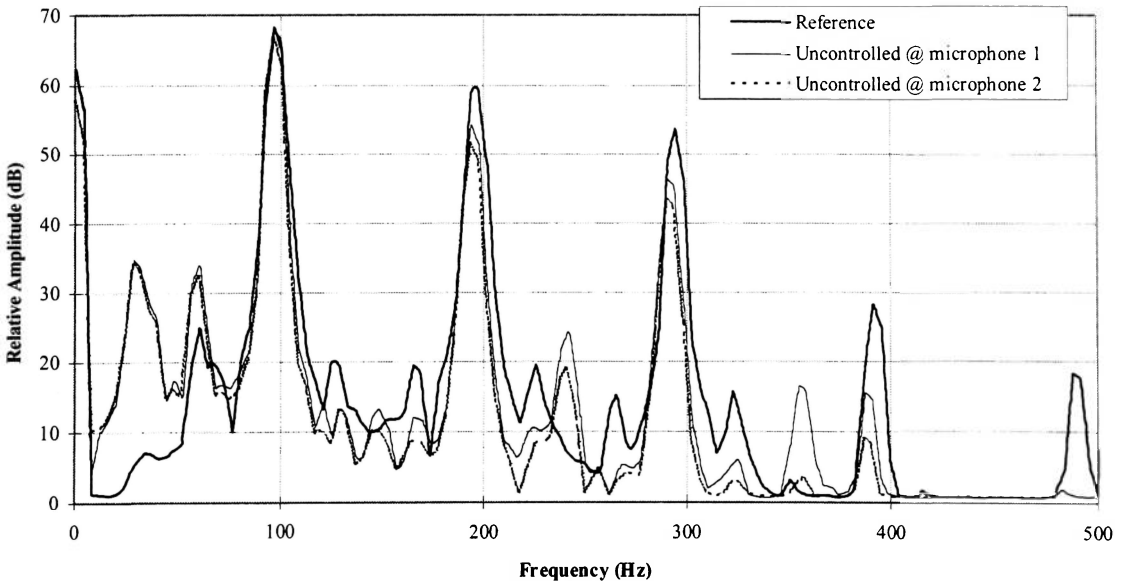


Figure 5.3.1a. Reference signal and cabin noise spectra at each of two feedback microphones

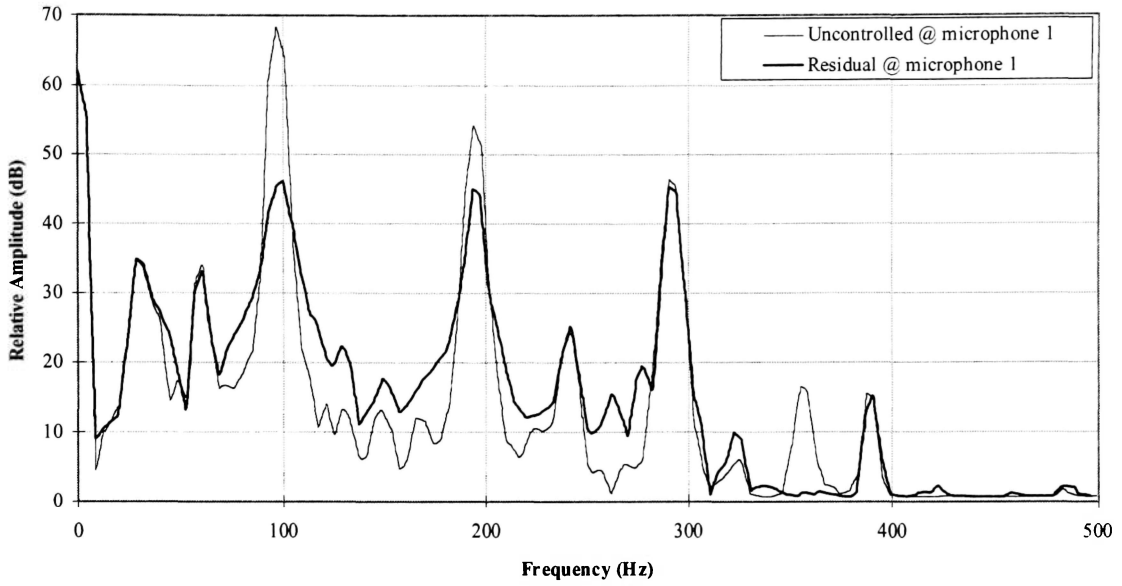


Figure 5.3.1b. Cabin noise spectrum at microphone number 1 before control and residual noise during control

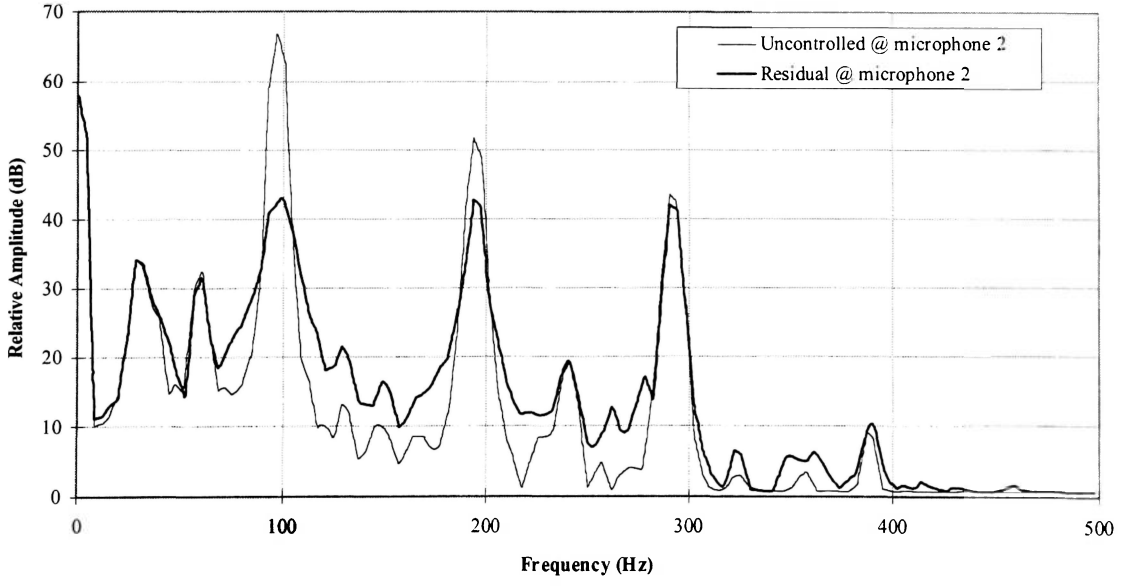


Figure 5.3.1c. Cabin noise spectrum at microphone number 2 before control and residual noise during control

The spatial extent of the reduced noise field through out the cabin is not evaluated for several reasons as describe earlier and summarized here. The acoustic response of the cabin enclosure has a large effect on the extent of the reduction throughout the cabin and each enclosure needs to be analyzed and surveyed to determine the optimum locations for the transducers. The scope of this research does not allow for an analysis of the acoustic response of the enclosure in which the system is tested. An experimental technique to survey the acoustic response of the enclosure is also beyond the scope of this work. Testing is not performed in a representative fuselage and the spatial extent of the reduced sound field in the testing enclosure is not representative of the results which can be achieved in a fuselage. The best result which can be hoped for is to achieve good local control which is used to show the capability of the control system. Testing in representative fuselages is necessary to obtain results which are representative of the spectral reductions possible and the spatial extent of the reduced sound field during flight.

CHAPTER 6
CONCLUSIONS

The purpose of this work is to build and test a notebook computer based active noise control (ANC) system to initiate investigation into the feasibility and the performance potential of active noise control to reduce cabin noise in single engine, general aviation aircraft. The "filtered x" least mean square (fxLMS) algorithm is implemented in a finite impulse response (FIR) filter structure and a simple, cheap, acoustic feedforward method is used to generate the reference signal. The system is tested in a laboratory setup with cabin noise and reference signal data recorded during flight in a Piper Malibu Mirage, Piper Cherokee 140, and a Cessna 172.

The feedforward method using windshield vibrations is shown to be too sensitive to the aircraft type and the resulting lack of coherence between the reference signal and the cabin noise prevents control of the Piper Malibu Mirage and Cherokee 140 cabin noise. The feedforward method allows a reduction in the overall sound pressure levels (OASPL) in the Cessna 172 by approximately 8 dB to 9 dB. Further reductions are not possible in the Cessna 172 due to a strong engine tone in the cabin noise which is absent from the reference signal.

To simulate the system performance given a reference signal which has strong coherence with the cabin noise, the recorded cabin noise is used as both the reference signal and the signal to drive the primary speaker exciting the cabin noise in the laboratory tests. These tests result in cabin noise reductions by approximately 12 dB to 15 dB in OASPL in both the Piper Cherokee 140 and the Cessna 172. Blade passage frequency (BPF) tones are reduced by over 20 dB in both cases and the remainder of the tones are reduced to the broadband level. Broadband reductions are not achieved. The results indicate the large reductions in cabin noise which are possible once the system is fully developed.

The current feedforward method is unlikely to yield the necessary coherence between the reference signal and the cabin noise. Alternative, tonal, reference signal sources such as those obtained from

the engine or propeller shaft are good candidates for future systems. To simulate the use of a tonal reference signal, a filtered ramp function is used to simulate the tones due to BPF and the harmonics of BPF. These tests show good control of the BPF tone and the first harmonic. Control of the second harmonic is minimal because the acoustic response of the testing enclosure reduces the coherence at the second harmonic. These tests show a tonal reference signals to be a feasible alternative for future study.

The results show the application of ANC to reduce cabin noise in single engine, general aviation aircraft to achieve significant noise reductions. The current feedforward method is shown to be unfeasible for application in a wide range of aircraft and large reductions are possible through the use of tonal reference signals taken from the engine or propeller. The presented work provides a solid base for future research and development work by additional graduate students.

CHAPTER 7
RECOMMENDATIONS

It is only a matter of time before ANC systems appear in single engine, GA aircraft. Several major issues must first be resolved, the primary issue being cost. The system must be designed such that it can be installed in different aircraft models with minimal modifications other than speaker and microphone considerations. Such a system is suitable for installation in a large selection of aircraft models which will allow for reduced costs by spreading the engineering costs over a larger number of aircraft.

Future work should concentrate on several major areas before attempting to test flight any system in an aircraft. The first major problem which must be solved is the design and testing of a feedforward method which will yield the required coherence with the cabin noise. The feedforward method must be sufficiently versatile that it can be implemented in different aircraft models, with either two or three bladed propellers, with only minor changes to the system. If the feedforward method needs extensive work and redesign before it can be implemented in different aircraft, or aircraft with a different number of propeller blades, then the cost of applying ANC will not be feasible.

The use of an accelerometer to provide a broadband reference signal which has the necessary coherence with the cabin noise will allow broadband reductions at lower frequencies where it is the most noticeable. A suitable location may be difficult to determine as it is with the acoustic reference signal. It may be possible to add a small, non-structural member to the aircraft structure whose vibration response has been tuned to improve the accelerometer data coherence with the cabin noise. Such a feedforward system seems to offer more problems than solutions.

The recommended method is to use the multiple, tonal reference method outlined in the "Reference Signal Considerations" section. Figure 7.1 is a simple block diagram to illustrate the method recommended to generate and use the multiple, tonal reference signals. The figure shows one

signal coming from a feedforward system. The feedforward signal must consist of a pulse train and the feedforward system can be a pulse generator as described under "Reference Signal Considerations" in Section 2.8 or the pulse source can be taken from the electric tachometer (as is used in the Malibu Mirage). A series of phase-locked-loop frequency multipliers (and dividers as needed) manipulate the pulse train to generate one pulse train corresponding in frequency to each dominant tone in the cabin noise (labeled "F M/D" in Figure 7.1). Voltage controlled oscillators (labeled "VCO" in the figure) convert the pulse train to tones of the same frequency as the pulse train. The resulting tones are used as reference signals to the fxLMS control system.

One reference signal is used for each tone and each tone is adjusted through the use of a separate controller. As described in Chapter 2., under ideal situations only 2 weights are needed to control a tone. In reality a few weights are necessary to control each tone but the amount of calculation performed will still be less than required with the current, broadband reference signal. Similar reductions in calculation requirements may be achieved with the secondary path models if a tone, or narrow band limited white noise, is used during the identification. Appendix A includes a Mathcad worksheet containing a SISO simulation illustrating the use of 3 reference signals and a single feedback signal.

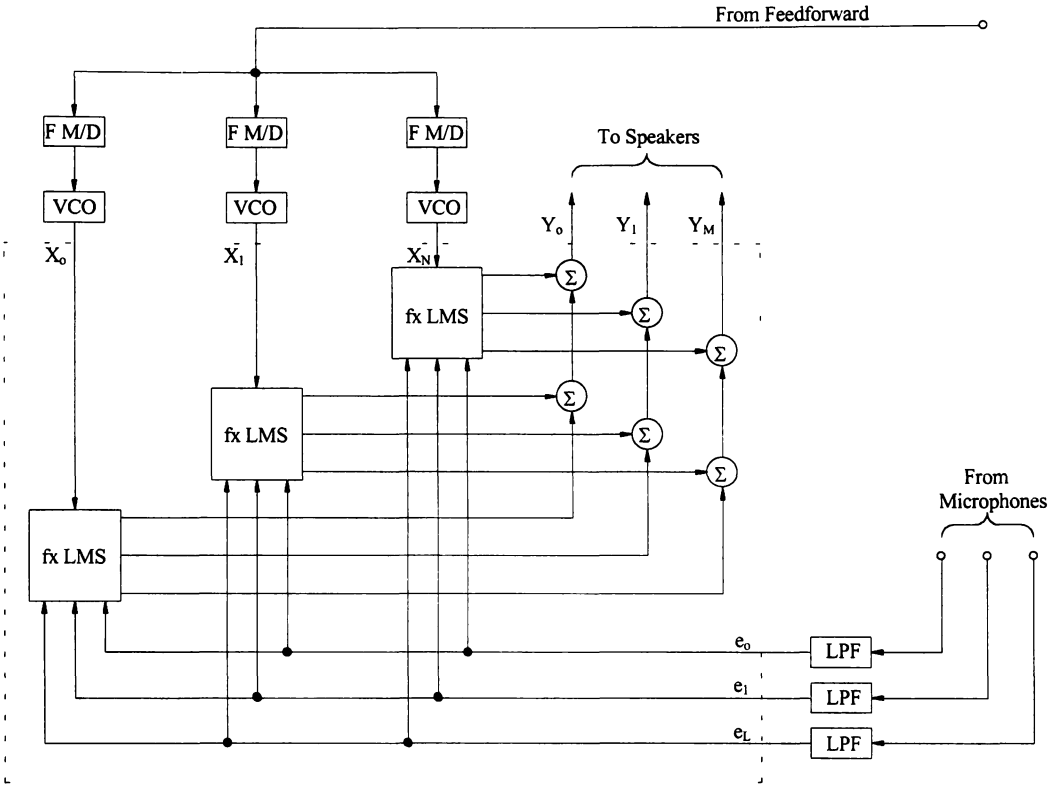


Figure 7.1 Block diagram to illustrate a multiple tonal reference method through the use of multiple fxLMS controllers

Figure 7.1 shows the same error signals are sent to each of the controllers instead of filtering the error signal such that each controller only receives the error signal corresponding to the tone it is controlling. The construction of narrow bandpass filters to filter the error signal to a very narrow frequency range will be very difficult and may be impossible. The math simulations in Appendix A indicate a broadband error signal can be used after lowpass filtering which greatly simplifies the filtering requirements to one lowpass filter for each microphone.

The benefit of a system using multiple tonal reference signals is that each tone is controlled independently of the other tones and coherence between the cabin tones is no longer an issue. It is necessary to ensure the phase difference between the reference signal and the cabin noise signal does not drift over time which is why phase-locked-loops are recommended for the frequency multipliers/dividers and the voltage controlled oscillators. Such a system also offers increased

flexibility because once the system identification is performed, continual or repeated identification during flight is probably not necessary although this must still be tested. Modifications to the feedforward system to accommodate different aircraft or a different number of propeller blades are relatively easily accomplished by changing the number of reference signals and/or the frequency multiplication/division factors and allowances can be built into the software to simplify the addition or removal of additional reference signals.

The main drawback to such a multiple, tonal reference signal is that some tones, such as the stray engine tone in the Cessna 172, may be difficult to include in the reference signal. If a dominant tone can not be constructed in the reference signal or picked up some other way, the transmission path of the tone should be investigated. It may be possible to eliminate this tone through a passive method such as a tuned vibration absorber suitably placed. The other drawback to this system is that broadband control is not possible.

An investigation into the cabin acoustics should be performed in parallel with the development of the algorithm and feedforward system. Due to a lack of aeroacoustic finite element modeling facilities at the university, simplified cabin modeling and analysis should be undertaken to develop a better understanding of the acoustic response of the cabin. An experimental acoustic survey of candidate aircraft should be performed to enable a determination of the best speaker and microphone locations, system implementation and the amount of global or local reduction that can be achieved. Such a survey is also necessary to determine the number of speakers and microphones needed since a minimal number of each is desired.

CHAPTER 8

REFERENCES

1. Richard, E. J. and Mead, D. J., *Noise and Acoustics Fatigue in Aeronautics*, Wiley, 1968.

2. Carlos Latoni, Chief Structural Engineer, Piper Aircraft, personal conversations, 1995 and 1996.

3. Hansen, C. H. and Snyder, S. D., *Active Control of Noise and Vibration*, E & FN Spon, 1997.

4. Lord NVX^{TM} Systems for Active Noise and Vibration Control, Lord Corporation, 1994.

5. Piper Saratoga II HP Cabin ANC System (confidential proposal to Piper Aircraft), Active Noise and Vibration Technologies, Inc., 1994.

6. Hubbard, H. H. ed., "Aeroacoustics of Flight Vehicles," NASA Reference Publication 1258, Vol. 1 & 2, 1991.

7. Patrick, H. V. L., "Cabin Noise Characteristics of a Small Propeller Powered Aircraft," AIAA 10th Aeroacoustics Conference, Seattle, WA, July 1986.

8. Cabell, R. H., Lester, H. C., Mathur, G. P., and Tran, B. N., "Optimization of Actuator Arrays for Aircraft Interior Noise Control," 15th AIAA Aeroacoustics Conference, Long Beach, CA, October 1993.

9. Fuller, C. R. and Gibbs, G. P., "Active Control of Interior Noise in a Business Jet using Piezoceramic Actuators," Noise-Con 94, Ft. Lauderdale, FL, May 1994.

10. Burdisso, R. A., Fuller, C. R., and Smith, J. P., "Control of Broadband Radiated Sound with Adaptive Structures," *SPIE Vol. 1917 Smart Structures and Intelligent Systems*, 1993.

11. Lord, H. W., Gatley, W. S., and Evensen, H. A., *Noise Control for Engineers*, Robert E. Krieger Publishing Co., 1987.

12. Elliott, S. J. and Nelson, P. A., "Active Noise Control," *Noise/News International*, June 1994.

13. Nelson, P. A. and Elliot, S. J., *Active Control of Sound*, Academic Press Limited, 1994.

14. Adachi, S., Kasuya, H., and Sano, H., "Application of Least Squares Lattice Algorithm to Active Noise Control for Automobile," Proceedings of the American Control Conference, Baltimore, Maryland, June 1994.

15. Koshigoe, A., Gordon, A., and Ndefo, E., "Comparison of Various Adaptive Control Algorithms," 45th Congress of the International Astronautical Federation, Jerusalem, Israel, October 1994.

16. Stearns, S. D. and Widrow, B., *Adaptive Signal Processing*, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1985.

17. Snyder, S. D. and Tanaka, N., "A Neural Network for Feedforward Controlled Smart Structures," *Journal of Intelligent Material Systems and Structures*, Vol. 4, July 1993.

18. Bendat, J. S. and Piersol, A. G., *Random Data: Analysis and Measurement Procedures*, John Wiley & Sons, 1986.

19. Morgan, D., *Practical DSP Modeling, Techniques, and Programming in C*, John Wiley & Sons, 1994.

APPENDIX A

MATH SIMULATIONS

SISO Identification Simulation 96

SISO LMS simulation 98

Spectral Analysis Tools 101

LMS ANC Simulation using 2 Reference Signals and 1 Feedback Signal 103

LMS ANC Simulation using 3 Reference Signals and 1 Feedback Signal 104

SISO Identification Simulation

$n = 0, 1 .. 7$

$Cmodel_n = \sin\left(\frac{n}{3}\right)$

$Cmodel_7 = 0$

Cmodel: FIR weights to simulate the secondary path response

$$fir(k, C, lengthC, X) = \sum_{n=0}^{lengthC-1} C_n \cdot X_{k-n}$$

The function $fir(k, C, lengthC, X)$ calculates the FIR filter output given:
k: time index value
C: FIR weight array name
lengthC: length of FIR weight array
X: name of input array

$PieceWiseID(length, X, gain) =$

$k \leftarrow 8$

for $repeat \in 0, 1 .. 7$

$C_{repeat} \leftarrow 0$

$out_{k+2,1} \leftarrow fir(k, Cmodel, 8, X)$

$out_{k,(4+repeat)} \leftarrow C_{repeat}$

$out_{k,0} \leftarrow X_k$

$k \leftarrow k + 1$

for $repeat \in 8, 9 .. length$

$out_{k,0} \leftarrow X_k$

$out_{k+2,1} \leftarrow fir(k, Cmodel, 8, X)$

$out_{k,2} \leftarrow fir(k, C, 8, X)$

$out_{k,3} \leftarrow out_{k,1} - out_{k,2}$

for $n \in 0, 1 .. 7$

$C_n \leftarrow C_n + gain \cdot out_{k,3} \cdot X_{k-n}$

$out_{k,(4+n)} \leftarrow C_n$

$k \leftarrow k + 1$

out

The function PieceWiseID(length , X, gain) performs the algorithm calculations which simulate the secondary path identification.

length: length of the reference signal array
X: name of reference signal array
gain: the gain value used in the LMS weight update algorithm

$gainID = 1 \cdot 10^{-6}$

outID = PieceWiseID(1000, RandomNoise , gainID)

The resulting ouput array from the simulation (outID) contains the reference signal, actual feedback signal, model feedback signal, error signal and the weight values. The simulation is called on by PieceWiseID(1000, RandomNoise , gainID) which performes the simulation through the use of a RandomNoise array of length 1000 with the gain as shown.

All results are postprocessed to evaluate the overall noise reduction in decibels and to evaluate the cabin noise spectrum before applying control and the residual cabin noise when control is applied. The "Spectral Analysis Tools" section contained in this section details the calculation tools used to perform the postprocessing of the simulation results

SISO LMS simulation

$junk := 0, 1 .. 20$

$W_{junk} := 0$

$W_0 := 1$

Control weight array (W) initialized to zero except the first value which is initialized to a value of one

$$fir(j, W, lengthW, X) := \sum_{i=0}^{lengthW-1} W_i \cdot X_{j-i}$$

The function $fir(k, C, lengthC, X)$ calculates the FIR filter output given:

j: time index value
W: FIR weight array name
lengthW: length of FIR weight array
X: name of input array

PieceWise(length, cabin, FeedFwd, lengthW, gain) :=

```
k ← 40
for repeat ∈ 0, 1 .. lengthW − 1
    out_{k,0} ← cabin_{k−2}
    out_{k,1} ← out_{k,0} + fir(k, W, lengthW, FeedFwd)
    out_{k,2} ← FeedFwd_k
    W_repeat ← 0
    k ← k + 1
W_0 ← 0.2
for repeat ∈ lengthW .. length
    out_{k,0} ← cabin_k
    out_{k,1} ← out_{k,0} + fir(k, W, lengthW, FeedFwd)
    out_{k,2} ← FeedFwd_k
    for n ∈ 0, 1 .. lengthW − 1
        W_n ← W_n − gain · out_{k,1} · FeedFwd_{k−n}
    k ← k + 1
out
```

The function PieceWise(length, cabin, FeedFwd, lengthW, gain) performs the algorithm calculations to simulate a SISO LMS control system.

length: length of the input array (containing the simulated noise)
cabin: name of the input array (containing the simulated noise)
FeedFwd: reference signal array name
lengthW: length of the controller FIR filter
gain: gain value used in the the LMS weight update algorithm

$$\text{toneNoise} := \left(\text{tone80}\cdot 2 + \text{tone160}\cdot 2 + \text{tone}\cdot 2 + \frac{\text{Noise}}{4} + \text{tone40}\cdot 2\right)\cdot 10($$

Noise signal constructed to consist of tones at 40 Hz, 80 Hz, 110 Hz and 160 Hz plus a random noise component

$$\text{lengthANC} := 120($$

$$\text{gain} := \frac{8}{\text{lengthW}}\cdot\frac{0.5}{\max(\text{feedFxS})\cdot\max(\text{cabin})}$$

$$\text{cabin} := \text{toneNoise}$$
$$\text{lengthW} := 20$$

Constructing reference signal which has strong coherence with cabin noise signal

$$\text{feedFxS} := \text{cabin}$$

Running SISO LMS simulation with the cabin noise constructed above and the reference signal which has high coherence with the cabin noise signal

$$\text{outStrong} := \text{PieceWise}(\text{lengthANC}, \text{cabin}, \text{feedFxS}, \text{lengthW}, \text{gain})$$

Constructing reference signal which has weak coherence with the cabin noise because it contains only two of the four tones in the cabin noise signal

$$\text{feedFxW}_q := \left(\text{tone80}_q\cdot 2 + \text{tone160}_q\cdot 2 + \frac{\text{Noise}_q}{2}\right)\cdot 10($$

Running SISO LMS simulation with the cabin noise constructed above and the reference signal which has low coherence with the cabin noise signal

$$\text{outWeak} := \text{PieceWise}(\text{lengthANC}, \text{cabin}, \text{feedFxW}, \text{lengthW}, \text{gain})$$

All results are postprocessed to evaluate the overall noise reduction in decibels and to evaluate the cabin noise spectrum before applying control and the residual cabin noise when control is applied. The "Spectral Analysis Tools" section contained in this Appendix details the calculation tools used to perform the postprocessing of the simulation results

Spectral Analysis Tools

The following calculations transform the an input array of time series data to the corresponding single-sided, frequency spectrum

$Nfft := 2^9$

Nfft: the number of data points used by the Fast Fourier Transform (FFT)

$i := 0, 1 .. Nfft - 1$

i: counter used to ensure there are Nfft data points in the input array used in the FFT calculations

Copying Nfft data points from the input array into the x array

$x_i := array_1$

$$mean_x := \sum_{n=0}^{Nfft-1} \frac{x_n}{Nfft - 2}$$

Calculating the mean value of all the data in the x array

$$x := \overrightarrow{(x - mean_x)}$$

Normalizing the x array such that it has zero mean

Calculating the mean-sqaure value of all the data in the x array

$$meanSQ_x := \sum_{n=0}^{Nfft-1} \frac{(x_n)^2}{Nfft - 2}$$

Calculating the autospectral density function for x. The resulting array (G_{XX}) is the spectrum of the time domain data contained in the array x

$$G_{xx} := \overrightarrow{(|fft(x_k)|)^2}$$

Expressing the spectrum gain in terms of (relative) Decibels

$$dbG_{xx} := \overrightarrow{\left[10 \cdot \log\left[\frac{|G_{xx}|}{(2 \cdot 10^{-5})^2}\right]\right]}$$

$$j = 0, 1 .. \frac{Nfft}{2}$$

j: counter used to evaluate the frequency

$$f_S = \frac{1}{700\,10^{-6}}$$

f_S: samping frequency (samples per second)

$freq_j$: frequency values corresponding to the dbG_{XX} calculated above

$$freq_j = \frac{f_S}{Nfft} \cdot j$$

Calculation of the noise reduction (in decibels) through the use of the mean-square value calculated for the uncontrolled noise before control is applied and the residual noise during control

Calculations to evaluate the frequency domain transfer function gain and phase from a given finite impulse response function in the form of an array:

$$resp = FFT(impulse)$$

Calculation of the complex frequency domain transfer function (resp) from the given impulse response array (impulse)

Calculation of the frequency domain transfer function gain from the complex transfer function (resp)

$$\overrightarrow{gain = (20 \cdot \log(|resp|))}$$

Calculation of the frequency domain transfer function phase from the complex transfer function (resp)

$$\overrightarrow{phase = (57.3 \cdot \arg(resp))}$$

LMS ANC Simulation using 2 Reference Signals and 1 Feedback Signal

$$
\begin{aligned}
&2ff(\text{length}, \text{cabin}, X1, X2, \text{lengthW}, \text{gain}) = \\
&\quad k \leftarrow 50 \\
&\quad \text{for } \text{repeat} \in 0, 1 .. \text{lengthW} \\
&\quad\quad out_{k,0} \leftarrow \text{cabin}_k \\
&\quad\quad W1_{\text{repeat}} \leftarrow 0 \\
&\quad\quad W2_{\text{repeat}} \leftarrow 0 \\
&\quad\quad out_{k,1} \leftarrow 0 \\
&\quad\quad out_{k,2} \leftarrow out_{k,1} + out_{k-2,0} \\
&\quad\quad k \leftarrow k + 1 \\
&\quad W_0 \leftarrow -0.5 \\
&\quad \text{for } \text{repeat} \in 8, 9 .. \text{length} \\
&\quad\quad out_{k,0} \leftarrow \text{cabin}_k \\
&\quad\quad y_k \leftarrow fir(k, W1, \text{lengthW}, X1) + fir(k, W2, \text{lengthW}, X2) \\
&\quad\quad out_{k+0,1} \leftarrow y_k \\
&\quad\quad out_{k,2} \leftarrow out_{k,1} + out_{k,0} \\
&\quad\quad out_{k,3} \leftarrow X1_k \\
&\quad\quad out_{k,4} \leftarrow X2_k \\
&\quad\quad \text{for } n \in 0, 1 .. \text{lengthW} \\
&\quad\quad\quad W1_n \leftarrow W1_n - \text{gain} \cdot out_{k,2} \cdot out_{k-n,3} \\
&\quad\quad\quad W2_n \leftarrow W2_n - \text{gain} \cdot out_{k,2} \cdot out_{k-n,4} \\
&\quad\quad k \leftarrow k + 1 \\
&\quad \text{for } n \in 0, 1 .. \text{lengthW} \\
&\quad\quad out_{k+n,0} \leftarrow W1_n \\
&\quad\quad out_{k+n,1} \leftarrow W2_n \\
&\quad out
\end{aligned}
$$

Calling the function $2ff(\text{length}, \text{cabin}, X1, X2, \text{lengthW}, \text{gain})$ runs the simulation for the given parameters:

length: length of the input noise and reference arrays
cabin: array containing cabin noise
X1, X2: arrays containing reference signals
lengthW: length of each FIR filter
gain: gain use in the LMS weight update algorithm

LMS ANC Simulation using 3 Reference Signals and 1 Feedback Signal

This simulation shows the use of three tonal reference signals (one tone per reference signal) to reduce three dominant tones in the cabin noise. The cabin noise is constructed to consist of broadband random noise and three tones which dominant over the broadband component by 20 dB. Each of the reference signals is acted on by its own controller independantly of the action taken on the other reference signals.

Notice the error signal is not filtered before it is used to update each of the weight arrays and the same, broadband error signal is used to update each of the three control weight vectors.

```
LMSanr3ff( length , cabin , X1, X2, X3, wlength , gain ) =
    k ← 10
    start ← 286
    for repeat ∈ 0, 1 .. start
        out_{k,0} ← cabin_k
        out_{k,1} ← 0
        out_{k,2} ← out_{k,1} + out_{k-2,0}
        k ← k + 1
    for repeat ∈ 0, 1 .. wlength
        W1_repeat ← 0
        W2_repeat ← 0
        W3_repeat ← 0
    W1_0 ← 0 1
    for repeat ∈ start + 1 .. length
        out_{k,0} ← cabin_k
        y_k ← fir( k, W1, X1, wlength ) + fir( k, W2, X2, wlength ) + fir( k, W3, X3 wlength )
        out_{k+1,1} ← y_k
        out_{k,2} ← out_{k,1} + out_{k,0}
        out_{k,3} ← X1_k
        out_{k,4} ← X2_k
        out_{k,5} ← X3_k
        for n ∈ 0, 1 .. wlength
            W1_n ← W1_n - gain out_{k,2} out_{k-n,3}
            W2_n ← W2_n - gain out_{k,2} out_{k-n,4}
            W3_n ← W3_n - gain out_{k,2} out_{k-n,5}
        k ← k + 1
    out
```

$q := 0, 1 \ldots 2000$ q: counter

$f_s := \frac{1}{700 \cdot 10^{-6}}$ f_S: sampling frequency

$tone40_q := \sin\left(2 \cdot \pi \cdot 41 \cdot \frac{q}{f_s}\right)$ tone40: constructed 40 Hz tone

$max40 := max(tone40)$ max40: maximum value of tone40

$tone80_q := (tone40_q)^2 \cdot 2 - max40($ tone80: constructed 80 Hz tone

$max80 := max(tone80)$ max80: maximum value of tone80

$tone160_q := (tone80_q)^2 \cdot 2 - max80($ tone160: constructed 160 Hz tone

$tone110_q := \sin\left(2 \cdot \pi \cdot 110 \cdot \frac{q}{f_s}\right)$ tone110: constructed 110 Hz tone

$Noise_q := \frac{rndNum_q}{300}$ Noise: scaled random noise

$toneNoise_q := (tone80_q \cdot 4 + tone160_q \cdot 4 + Noise_q + tone110_q \cdot 3) \cdot 10($ toneNoise: constructed cabin noise

$length := 190($ length of simulation

$CabinNoise := toneNoise$ constructing cabin noise to consist of tones and broadband noise

$ff1 := tone80 \cdot 10($
$ff2 := tone160 \cdot 20($
$ff3 := tone110 \cdot 20($ ff1, ff2, ff3: 3 tonal reference signals

$wlength := 3$ wlength: length of FIR filter applied to each reference signal

$gain := \frac{3}{max(CabinNoise)^2}$ gain used to in LMS weight update algorithm

$num := 10, 11 \ldots length$ num: counter used to plot time domain results

$time_{num} = \frac{num}{f_s}$ time: converted from sample number to time in seconds

out3ff =LMSanr3ff(length ,CabinNoise ,ff1 ,ff2 ,ff3 ,wlength ,gain)

out3ff calls the simulation and writes the results to out3ff array

The following illustrate the time and frequency domain results: Control is applied after 500 samples are passed (at approximately 0.2 seconds). Figure A1 shows the very rapid convergence to the minimal error state and provides a visual comparison or the amount of reduction.

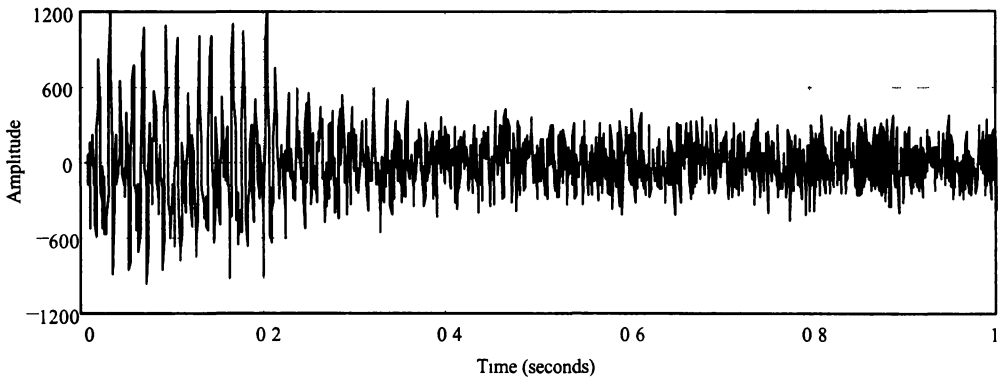


Figure A1. Simulated Cabin Noise

$Nfft = 2^{10}$ $Nfft = 1.024 \cdot 10^3$ Nfft: specifying 1024 points for use in FFT

$i = 0, 1 .. Nfft - 1$ i: counter

start = rows(out3ff)− wlength − Nfft start: calculating starting point from which to use data for use in FFT

$Before_i = out3ff_{i+start,0}$

$After_i = out3ff_{i+start,2}$

Before, After: arrays to store the last 1024 points of time domain results for use in the FFT to calculate the spectrum before and after (during) control

$ref1_i = ff1_i$

$ref2_i = ff2_i$

$ref3_i = ff3_i$ Arrays containing reference signal data for use in the FFT calculations

$$before = \overrightarrow{\left(|fft(Before)|\right)^2}$$

$$after = \overrightarrow{\left(|fft(After)|\right)^2}$$

before, after: arrays to store intermediate results

$$Ref_1 = \overrightarrow{\left[10\cdot\log\left[\left(\frac{|fft(ref1)|}{2\cdot10^{-5}}\right)^2\right]\right]}$$

$$Ref_2 = \overrightarrow{\left[10\cdot\log\left[\left(\frac{|fft(ref2)|}{2\cdot10^{-5}}\right)^2\right]\right]}$$

$$Ref_3 = \overrightarrow{\left[10\cdot\log\left[\left(\frac{|fft(ref3)|}{2\cdot10^{-5}}\right)^2\right]\right]}$$

Calculating spectrum for each of the reference signals

$$dBbefore = \overrightarrow{\left(10\cdot\log\left(\frac{|before|}{2\cdot10^{-5}}\right)\right)}$$

$$dBafter = \overrightarrow{\left(10\cdot\log\left(\frac{|after|}{2\cdot10^{-5}}\right)\right)}$$

dBbefore, dBafter: arrays containing the (relative) spectral amplitude results

$$j = 0, 1 .. \frac{Nfft}{2} \qquad freq_j = \frac{f_s}{Nfft} j$$

freq: calculating frequency data

Figure A2 shows the cabin noise spectrum and the spectrum of each of the three reference signals. Note from the figure the difference in amplitude between any reference signals and the cabin noise signal. As long as the reference signal frequency and phase does not drift with respect to the cabin noise, the algorithm is capable of reducing each of the tones present in the reference signals as illustrated in Figure A3. From Figure A3 notice the tones are reduced but there is no broadband reduction.

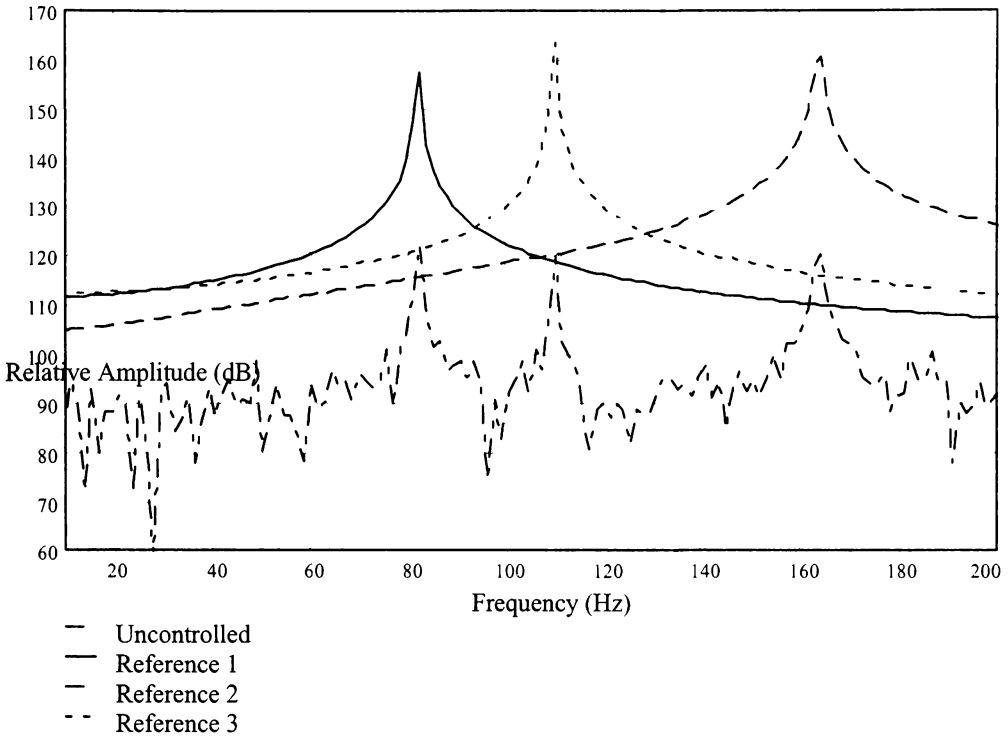


Figure A2. Reference and Cabin Spectra

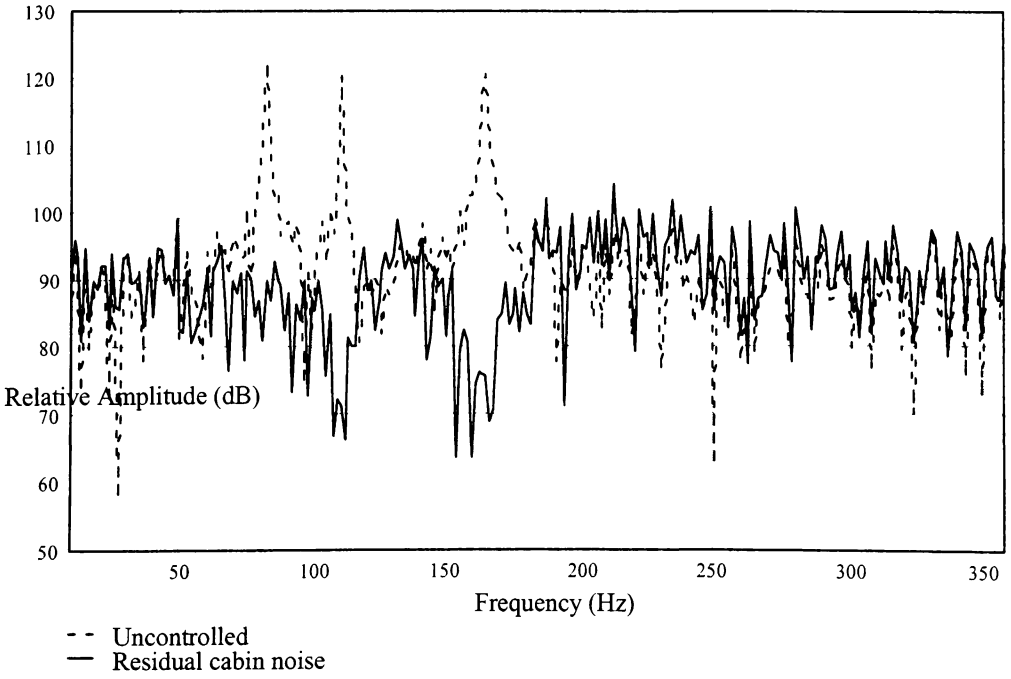


Figure A3. Cabin Noise Spectral Results

APPENDIX B

EQUIPMENT LISTING

Description	Quantity
Advent model Prodigy Tower II audio speakers	2
Austin Direct 80486 100 MHz laptop computer	1
Austin Direct expansion docking station EDS20SN	1
B&K model 2230 Precision Integrated Sound Level Meter	1
Hewlett Packard model 3310A function generator	1
Hewlett Packard model 3582A Spectrum Analyzer	1
JVC model UX-1 Micro Component System	1
Krohn-Hite model 3340 (high and high pass) filter	2
National Instruments Corp. model AT-AO-6 A/D board	1
National Instruments Corp. model AT-MIO E-10 data acquisition board	1
National Semiconductor model LM336 Precision Voltage Regulator	1
Panasonic Electret Condenser Microphone Cartridge Part Number WM-034CY 195	2
Realistic model SCT-24A stereo cassette tape deck	1
Realistic model STA-115 AM/FM stereo receiver and amplifier	1
Sony Tektronix model 314 Storage Oscilloscope	1

APPENDIX C

'C' CODE LISTINGS

SISO Identification Program Listing 111

SISO Active Noise Control Program Listing 119

SISO Identification Program Listing

```
/*************************************************************************
ID.C                                                written by Jeroen H. Dolmans

This program works with the NI AT-AO-6 D/A board and the NI AT-MIO E-10 A/D board to perform the
system identification. The boards operate in DMA mode and the driver software is included, and called,
through the use of the following libraries:

                        #include "nidaq.h"
                        #include "nidaqerr.h"
                        #include "nidaqcns.h"

This program is for the SISO system only.
*************************************************************************/

/*  including standard libraries and NI-DAQ driver libraries */
#include <malloc.h>
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include <conio.h>
#include <fcntl.h>
#include <sys\types.h>
#include "nidaq.h"
#include "nidaqerr.h"
#include "nidaqcns.h"

/*declaration of CONTROL functions*/
 void get_data(void);
 void out(void);
 void fir(void);
 void LMS(void);
 void fx(void);

/*declaration of CONTROL global variables */
 unsigned int     length = 60, Clength = 60, i, block, halfSize;
 unsigned long int          junk=747, newestPtIndex, repeat=0;
 float    *weight, *gain, *W, *C, *cHat, *endRefX, *fX, *XRefLMS, *XRefC, *filteredX,
          *error, *errorEnd;
 double  MSE=0, MSX=0,ratioXtoE;
 int      *history, *data, *dataRef, *endAdd, *output, *y, *CdataRef, *errRef,*desired;

/*declaration of BOARD global variables*/
int       *inCircularBuffer,    /*circular buffer for DAQ double buffering*/
          *outCircularBuffer;   /*circular buffer for D/A double buffering*/

int       status,                          /*returned by NI-DAQ function calls*/
          InDeviceNumber = 1,              /*MIO-16E-10 device number assigned by NI-DAQ*/
          OutDeviceNumber = 2,             /*AT-AO-6 device number assigned by NI-DAQ*/
          inNumChans = 2,                  /*number of input channels*/
          outNumChans = 2,                 /*number of output channels*/
```

```
        outChanVect[2]= {0, 1},          /*vector of channels*/
        daqStopped,
        OutReady = 0;    /*OutReady = 1 when OutReady for next half buffer*/

unsigned long    HalfSize = 2; /*number of points in halfBuffer*/

/*declaration of BOARD functions*/
void    ErrClean(char *, int), /*error message, clean up and shut down function*/
        InConfig(void),          /*functions to configure the boards*/
        OutConfig(void),
        InEnding(void),          /*functions to close, cleanup and end*/
        OutEnding(void);

/*assigning file pointer for output file*/
 FILE *outfile;


/*  Starting main program  */
void main(){
        halfSize = (int)HalfSize;

 printf("enter # of model weights (eg: 16):  ");
 scanf("%d", &Clength);
 length = Clength;

/*assigning memory to pointers and initializing arrays to zero*/
 inCircularBuffer = (int*) calloc ((4*halfSize), sizeof(int));
 if(!inCircularBuffer){
        printf("\n unable to allocate history array\n");
        exit(1);
 }

 outCircularBuffer = (int *) calloc ((2*halfSize), sizeof(int));
 if(!outCircularBuffer){
        printf("\n unable to allocate history array\n");
        exit(1);
 }

 history = (int*) calloc ((length*2*halfSize), sizeof(int));
 if(!history){
        printf("\n unable to allocate history array\n");
        exit(1);
 }

 output = (int*) calloc (halfSize, sizeof(int));
 if(!output){
        printf("\n unable to allocate output array\n");
        exit(1);
 }

 filteredX = (float*) calloc ((length*2*halfSize), sizeof(float));
 if(!filteredX){
        printf("\n unable to allocate filteredX\n");
        exit(1);
```

```
  }

  C = (float*) calloc (Clength, sizeof(float));
  if(!C){
          printf("\n unable to allocate C array\n");
          exit(1);
  }

  weight = (float*) calloc (length, sizeof(float));
  gain = (float*) calloc (1, sizeof(float));
  error = (float*) calloc (1, sizeof(float));
  desired = (int*) calloc (1, sizeof(int));

  if(!weight || !gain || !error || !desired){
          printf("\n unable to allocate weight or gain or error or desired\n");
          exit(1);
  }

 /*  initializing pointers */
  W = weight;
  *W = 1;
  y = output;
  desired = history + 1;
  data = history;
  dataRef = history;
  endAdd = history + (length*2*halfSize) - 1;
  cHat = C;
  endRefX = filteredX + (length*2*halfSize) - 1;
  fX = filteredX;
  errorEnd = error;

  /* prompting user for input*/
  printf("enter gain (eg 3e-7):  ");
  scanf("%e", gain);
  clrscr();
  printf("gain = %e\n", *gain);


 /*configuring and starting boards*/
  status = USE_E_Series_DAQ();
  if (status) ErrClean("USE_E_Series_DAQ()", status);

  status = USE_AO_610();
  if (status) ErrClean("USE_AO_610", status);

  OutConfig();

  InConfig();
  printf("both boards configured and started\n");
  printf("entering identification loop\n");

 /* STARTING UP THE IDENTIFICATION ALGORITHM*/
 /*setting up loop to repeatedly fill buffers and empty buffers*/
  while (!kbhit()) {
```

```
            get_data();
            if(filteredX == endRefX) filteredX = fX;
            else filteredX++;

            fx();
            fir();
            LMS();

            if(ratioXtoE > 1.5e5) break;

            dataRef += inNumChans;
            if (dataRef > endAdd) dataRef = history;
            data = dataRef;
  }

 /*  writing identification weights to file  and closing file*/
  printf("MSE = %.2lf, MSX = %.2lf,  MSX/MSE = %.2lf\n", MSE, MSX, MSX/MSE);
  outfile = fopen ("c-wght.dat", "wt");
  if (outfile == NULL){
            printf("\nunable to open file for output\n");
            exit(1);
  }

  for (i = 0; i < Clength; i++){
            fprintf(outfile,"%f\n",*C++);
  }

  fclose(outfile);

/*  terminating program  */
  InEnding();
  OutEnding();
  ErrClean("DONE",0);

  //printf("gain = %e, HalfSize = %d\n", *gain, HalfSize);
  //printf("filter length = %d\n repeat value = %d\n", Clength, repeat);
}
/* end of main program */


/*************************************************************************
InConfig():       function to configure and set up AT-MIO-16E10 board to output data
                  at 1200 samples per second from file named above
*************************************************************************/
void InConfig(void)
{
/* declaration of variables */
  int       mode = 1,                   /*mode*/
            i = 0,                      /*index*/
            inGainVect[2] = {20,20}, /*gain vector */
            inChanVect[2]= {0, 1};    /*vector of channels*/

unsigned long     count = 8L*HalfSize;  /*number of points in circularBuffer*/
```

```
/*configuring input device*/
        for(i = 0; i<inNumChans; i++){
                status = AI_Configure(InDeviceNumber, i, 0, 20, 0, 0);
                if (status) ErrClean("AI_Configure", status);
        }

 status = DAQ_StopTrigger_Config(InDeviceNumber, 0, 0L);
 if (status) ErrClean("DAQ_StopTrigger_Config", status);

 status = DAQ_DB_Config (InDeviceNumber, mode);
 if (status) ErrClean("WFM_DB_Config", status);

/*setting up channels*/
 status = SCAN_Setup (InDeviceNumber, inNumChans, inChanVect, inGainVect);
 if (status) ErrClean("SCAN_Setup", status);

/*starting board*/
/*A/D board*/
 status = SCAN_Start(InDeviceNumber, inCircularBuffer, count, 1,
                                        10, 1, 700);
/*end of function*/
}

/*****************************************************************************
 InEnding():      function to terminate collecting data and cleaning up DAQ device
 *****************************************************************************/
 void InEnding(void)
 {
 status = DAQ_Clear(InDeviceNumber);
 ErrPrint("DAQ_Clear", status);

 }

/*****************************************************************************
OutConfig():      function to configure and set up AT-AO-6 board to output data
*****************************************************************************/
void OutConfig(void)
{
 int     sigCode = 0,                    /*indicating OUT0* signal for group 1 channels*/
         trigline = 0,                   /*indicating RTSI 0*/
         dir = 1;                        /*indicating acting as source for RTSI 0*/

/* driving Out0* signal onto RTSI 0, ie acting as source*/
 status = RTSI_Conn(OutDeviceNumber, sigCode, trigline, dir);
 if (status) ErrClean("RTSI_Conn", status);

/* using one shot update mode*/
        status = AO_Configure(OutDeviceNumber, 0, 0, 1, 2.5, 1);
        if(status) ErrClean("AO_Configure", status);

/*end of function*/
}

/*****************************************************************************
```

```
 OutEnding():     function to terminate sending out data, and cleaning up D/A device
*************************************************************************/
 void OutEnding(void)
 {
/*  status = WFM_Group_Control(OutDeviceNumber, 1. 0);
  if (status) ErrClean("WFM_Group_Control", status);                    */

  status = RTSI_Clear(OutDeviceNumber);
  if (status) ErrClean("RTSI_Clear", status);

 }


/*************************************************************************
        void ErrClean(proc_name, err)

        This function prints the error,if any, from a NI-DAQ DOS function.
        In case of an error memory allocated for buffer is freed and the
        program is ended.
                'proc_name': string of the name of the NI-DAQ DOS function
                'err': the error returned from the NI-DAQ DOS function
                'value_buf': space for binary values read from input channel
*************************************************************************/
void ErrClean(char *procname,int err)
{
        ErrPrint(procname,err);
        if (err != noErr)
        {
                if (inCircularBuffer != NULL) hfree(inCircularBuffer);
                if (data     != NULL) hfree(data);
                if (output     != NULL) hfree(output);
                if (outCircularBuffer != NULL) hfree(outCircularBuffer);
                printf("\nSorry! Exiting program because of the error\n");
                printf("MSE = %.2lf, MSX = %.2lf,  MSX/MSE = %.2lf\n", MSE, MSX, MSX/MSE);

                exit(1);
        }
 hfree(inCircularBuffer);
 hfree(history);
 hfree(error);
 hfree(cHat);
 hfree(weight);
 hfree(gain);
 hfree(fX);
 hfree(output);

}


/*************************************************************************/
/*FUNCTION get_data():
        This function reads values from the A/D process and places them into the
        array pointed to by the data pointer
        This function also sends the output data to the D/A process using the
```

```
        array pointed to by the output pointer
*************************************************************************/
void get_data(){
        output = y;

/* writing new values to output board   */
        status = AO_Write(OutDeviceNumber, 0, *output);
                if (status) ErrClean("AO_Write #0", status);
        status = AO_Write(OutDeviceNumber, 1, (int)*error);

// updating dac, i.e. converting new values to voltage outputs
        status = AO_Update(OutDeviceNumber);
                if (status) ErrClean("AO_Update", status);

/* getting next sequential block of data */
        status = DAQ_Monitor(InDeviceNumber, -1, 0, inNumChans, data, &newestPtIndex,
                                                                &daqStopped);
        if (status) ErrClean("DAQ_Monitor***", status);

/*  checking data to ensure isn't old data */
        while (junk == newestPtIndex){
          status = DAQ_Monitor(InDeviceNumber, -1, 0, inNumChans, data, &newestPtIndex,
                                                                &daqStopped);
          if (status) ErrClean("DAQ_Monitor***", status);
        }
        junk = newestPtIndex;

/*  calculating next randowm data point  */
        *data = random(1000) - 500;
        desired = data + 1;
}
/*************************************************************************/
/*FUNCTION fir():
        This function applies a FIR filter (weight vector C) to the array of reference
        data (data array)
/*************************************************************************/

void fir(){
 *output = (int)((*data) * (*W));
 for (i = 1; i < length; i++){
          if (data == history) data = endAdd - 1;
          else data -= 2;
          W++;
          *output += (int)((*data) * (*W));
 }
 //printf("o %d\n", *output);
 data = dataRef;
 output+= inNumChans;
 W = weight;
}

/*************************************************************************/
/*FUNCTION LMS():
        This function applies the LMS algorithm to the weight
```

```
        array using the given data and the error data
/*************************************************************************/
void LMS(){
 *error = (float)*desired - *filteredX;

 MSE = (*error * *error - MSE)/50;
 MSX = ((double)*desired * (double)*desired - MSX)/50;
 ratioXtoE = MSX/MSE;

 for (i = 0; i < Clength; i++){
        *C +=((*gain) * (*data) * (*error));
        if (data == history) data = endAdd - 1;
        else data -= 2;
        C++;
 }
 C = cHat;
}

/*************************************************************************/
/* FUNCTION fx():
                this function filters the reference signal and is the fx in the fxLMS
                algorithm.  An FIR model of the secondary path is used and the weight
                values are contained in the C array.  cHat contains the premanent
                address to the start of the C array
/*************************************************************************/
void fx(){
 CdataRef = data;
 *filteredX = (*data) * (*C);

 for (i = 1; i < Clength; i++){
        C++;
        if (data == history) data = endAdd - 1;
        else data -= 2;
        *filteredX += (*data) * (*C);
 }
 C = cHat;
 data = CdataRef;
 //fprintf(outfile,"fX %d,   %p\n",*filteredX, filteredX);
}

/*************************************************************************/

/*  END OF ID PROGRAM FUNCTIONS  */

/*************************************************************************/
```

SISO Active Noise Control Program Listing

```
/*****************************************************************************
ANR.C                                                    written by Jeroen H. Dolmans

This program works with the NI AT-AO-6 D/A board and the NI AT-MIO E-10 A/D board to perform
active noise reduction.  The boards operate in DMA mode and the driver software is included, and called,
through the use of the following libraries:

                        #include "nidaq.h"
                        #include "nidaqerr.h"
                        #include "nidaqcns.h"

This program is for the SISO system only.
*****************************************************************************/

/*  including standard libraries and NI-DAQ driver libraries */
#include <malloc.h>
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include <conio.h>
#include <fcntl.h>
#include <sys\types.h>
#include "nidaq.h"
#include "nidaqerr.h"
#include "nidaqcns.h"

/*declaration of CONTROL functions*/
 void get_data(void);
 void out(void);
 void fir(void);
 void LMS(void);
 void fx(void);

/*declaration of CONTROL global variables */
 unsigned int length = 60, Clength = 60, i, block, halfSize;
 unsigned long int junk=99e9, newestPtIndex, repeat=0;
 float *weight, *gain, *W, *C, *cHat, *endRefX, *fX,*XRefLMS, *XRefC,*filteredX, *outAccum;
 double  MSE=0, MSX=0;
 int *history, *data, *dataRef, *endAdd, *output, *y, *CdataRef, *errRef, *error, *errStart, *x, *xStart,
        *anti, *antiStart;

/*declaration of BOARD global variables*/
int     *inCircularBuffer,    /*circular buffer for DAQ double buffering*/
        *outCircularBuffer;   /*circular buffer for D/A double buffering*/

int     status,                         /*returned by NI-DAQ function calls*/
        InDeviceNumber = 1,             /*MIO-16E-10 device number assigned by NI-DAQ*/
        OutDeviceNumber = 2,            /*AT-AO-6 device number assigned by NI-DAQ*/
        inNumChans = 2,                 /*number of input channels*/
        outNumChans = 2,                /*number of output channels*/
        outChanVect[2]= {0, 1},         /*vector of channels*/
```

```
        daqStopped,
        OutReady = 0;     /*OutReady = 1 when OutReady for next half buffer*/

unsigned long     HalfSize = 2; /*number of points in halfBuffer*/


/*declaration of BOARD functions*/
void      ErrClean(char *, int), /*error message, clean up and shut down function*/
          InConfig(void),              /*functions to configure the boards*/
          OutConfig(void),
          InEnding(void),              /*functions to close, cleanup and end*/
          OutEnding(void);
/*assigning file pointer for input file*/
 FILE   *outfile, *infile;


/*  Starting main program  */
void main(){
/*  reading identification weights from file  */
          infile = fopen ("c-wght.dat", "r");
          if (infile == NULL){
                    printf("\nunable to open file for reading secondary path weights");
                    exit(1);
          }
/*  opening file used to store control weights at end of program  */
 outfile = fopen ("W-wght.dat", "wt");
 if (outfile == NULL){
           printf("\nunable to open file for output\n");
           exit(1);
 }

/*  requesting input from user  */
 halfSize = (int)HalfSize;
 printf("enter # of anc weights (eg: 16):  ");
 scanf("%d", &length);
 printf("enter # of model weights (eg: 16):  ");
 scanf("%d", &Clength);

/*assigning memory to pointers and initializing arrays to zero*/
 inCircularBuffer = (int*) calloc ((4*halfSize), sizeof(int));
 if(!inCircularBuffer){
          printf("\n unable to allocate history array\n");
          exit(1);
 }
 //printf("inCircularBuffer = %p",inCircularBuffer);

history = (int*) calloc (length*2*halfSize, sizeof(int));
 if(!history){
          printf("\n unable to allocate history array\n");
          exit(1);
 }
 //printf("history = %p  ",history);

 output = (int*) calloc (halfSize, sizeof(int));
```

```
  if(!output){
          printf("\n unable to allocate output array\n");
          exit(1);
  }
  //printf("   output = %p ", output);

  filteredX = (float*) calloc (length*2*halfSize, sizeof(float));
  if(!filteredX){
          printf("\n unable to allocate filteredX\n");
          exit(1);
  }
  //printf("  filteredX = %p\n", filteredX);

  C = (float*) calloc (Clength, sizeof(float));
  if(!C){
          printf("\n unable to allocate C array\n");
          exit(1);
  }
  //printf("C = %p  ",C);

  weight = (float*) calloc (length, sizeof(float));
  //printf("   weight = %p",weight);
  gain = (float*) calloc (1, sizeof(float));
  //printf("  gain = %p\n",gain);
  error = (int*) calloc (9000, sizeof(int));
  x = (int*) calloc (9000, sizeof(int));
  anti = (int*) calloc (9000, sizeof(int));
  //printf("error = %p",error);

  if(!weight || !gain || !error || !x || !anti){
          printf("\n unable to allocate weight or gain or error\n");
          exit(1);
  }

/*  initializing pointers */
  W = weight;
  y = output;
  data = history;
  dataRef = history;
  endAdd = history + length*2*halfSize - 1;
  cHat = C;
  endRefX = filteredX + length*2*halfSize - 1;
  fX = filteredX;
  errStart = error;
  xStart = x;
  antiStart = anti;

//reading secondary path weights from file*/
          for (i = 0; i < Clength; i++){
                  fscanf(infile, "%f", C);
                  C++;
          }
          C = cHat;
```

```
/* prompting user for input*/
 printf("enter gain (eg 3e-10):  ");
 scanf("%e", gain);
 clrscr();
 printf("gain = %e\n", *gain);

/*configuring and starting boards*/
 status = USE_E_Series_DAQ();
 if (status) ErrClean("USE_E_Series_DAQ()", status);

 status = USE_AO_610();
 if (status) ErrClean("USE_AO_610", status);

 OutConfig();
 InConfig();

/*STARTING UP THE CONTROL */
/*setting up loop to repeatedly fill buffers and empty buffers*/
 filteredX = endRefX;
 while (!kbhit()) {
          get_data();
          if(filteredX == endRefX) filteredX = fX;
          else filteredX++;

          fx();
          fir();
          LMS();

          dataRef += inNumChans;
          if (dataRef > endAdd) dataRef = history;
          data = dataRef;

 }

 printf("out of loop\n");

/*  terminating program  */
 InEnding();
 OutEnding();

/*  writing control weights to file  and closing file*/
 for (i = 0; i<length; i++){
          fprintf(outfile,"%f\n",*W);
          W++;
 }
 fclose(outfile);

/*  writting data to screen */
 printf("MSE = %.2lf, MSX = %.2lf, MSX/MSE = %.2lf\n", MSE, MSX, MSX/MSE);
 printf("gain = %e, HalfSize = %d\n", *gain, HalfSize);
 printf("filter length = %d\n", length);

 ErrClean("DONE",0);
```

```
}

/* end of main program */

/*************************************************************************
InConfig():     function to configure and set up AT-MIO-16E10 board to output data
                at 1200 samples per second from file named above
*************************************************************************/
void InConfig(void)
/* declaration of variables */
{
 int     mode = 1,                          /*mode*/
         i = 0,                             /*index*/
         inGainVect[2] = {20,20},           /*gain vector */
         inChanVect[2]= {0, 1};             /*vector of channels*/

 unsigned long   count = 8L*HalfSize;       /*number of points in circularBuffer*/

/*configuring input device*/
         for(i = 0; i<inNumChans; i++){
                 status = AI_Configure(InDeviceNumber, i, 0, 20, 0, 0);
                 if (status) ErrClean("AI_Configure", status);
         }

 status = DAQ_StopTrigger_Config(InDeviceNumber, 0, 0L);
 if (status) ErrClean("DAQ_StopTrigger_Config", status);

 status = DAQ_DB_Config (InDeviceNumber, mode);
 if (status) ErrClean("WFM_DB_Config", status);

/*setting up channels*/
 status = SCAN_Setup (InDeviceNumber, inNumChans, inChanVect, inGainVect);
 if (status) ErrClean("SCAN_Setup", status);

/*starting board*/
 status = SCAN_Start(InDeviceNumber, inCircularBuffer, count, 1, 10, 1, 700);

/*end of function*/
}

/*************************************************************************
 InEnding():     function to terminate collecting data and cleaning up DAQ device
 *************************************************************************/
 void InEnding(void)
 {
  status = DAQ_Clear(InDeviceNumber);
  ErrPrint("DAQ_Clear", status);
 }

/*************************************************************************
OutConfig():     function to configure and set up AT-AO-6 board to output data
*************************************************************************/

void OutConfig(void)
```

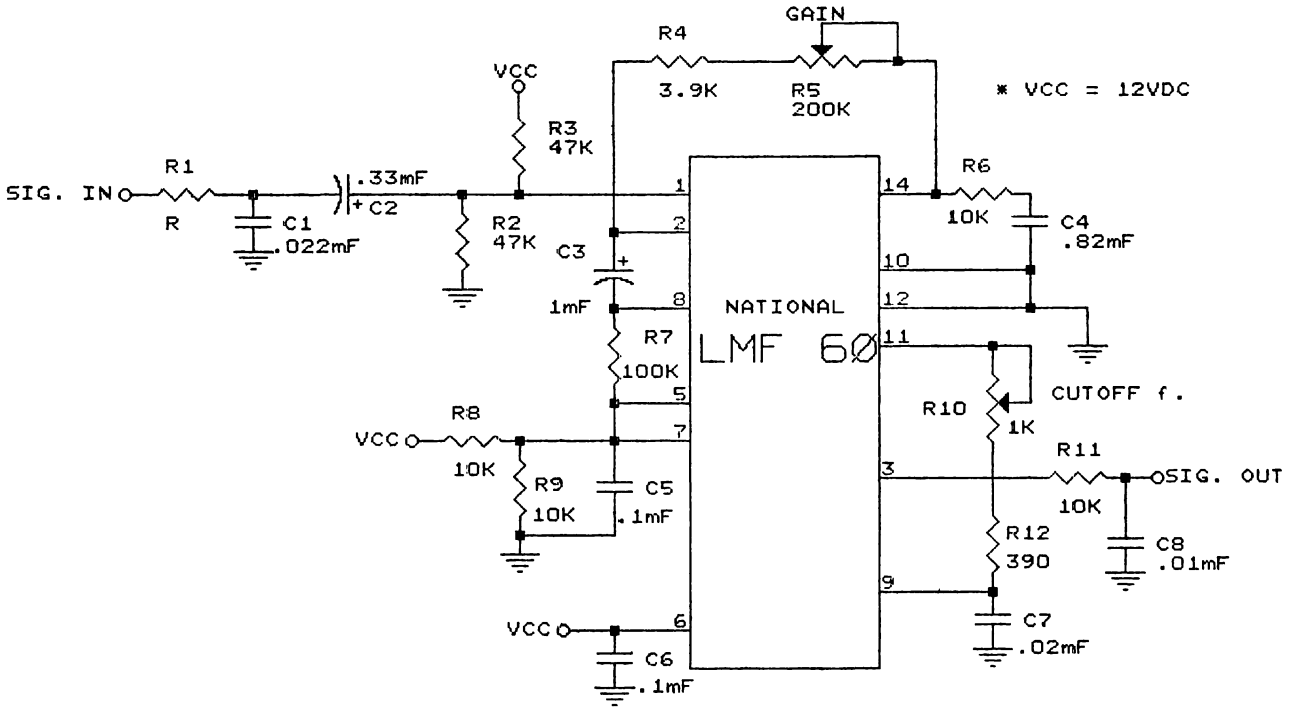
```
{
/*  declaration of variable  */
 int      sigCode = 0,              /*indicating OUT0* signal for group 1 channels*/
          trigline = 0,             /*indicating RTSI 0*/
          dir = 1;                  /*indicating acting as source for RTSI 0*/

/* driving Out0* signal onto RTSI 0, ie acting as source*/
 status = RTSI_Conn(OutDeviceNumber, sigCode, trigline, dir);
 if (status) ErrClean("RTSI_Conn", status);

/* using one shot update mode*/
          status = AO_Configure(OutDeviceNumber, 0, 0, 1, 2.5, 1);
          if(status) ErrClean("AO_Configure", status);

/*end of function*/
}

/*****************************************************************************
 OutEnding():     function to terminate sending out data, and cleaning up D/A device
*****************************************************************************/
 void OutEnding(void)
 {
 status = RTSI_Clear(OutDeviceNumber);
 if (status) ErrClean("RTSI_Clear", status);
}


/*****************************************************************************
          void ErrClean(proc_name, err)

          This function prints the error,if any, from a NI-DAQ DOS function.
          In case of an error memory allocated for buffer is freed and the
          program is ended.
                    'proc_name': string of the name of the NI-DAQ DOS function
                    'err': the error returned from the NI-DAQ DOS function
                    'value_buf': space for binary values read from input channel
*****************************************************************************/
void ErrClean(char *procname,int err)
{
          ErrPrint(procname,err);
          if (err != noErr)
          {
                    if (inCircularBuffer != NULL) hfree(inCircularBuffer);
                    if (data     != NULL) hfree(data);
                    if (output     != NULL) hfree(output);
                    if (outCircularBuffer != NULL) hfree(outCircularBuffer);
                    printf("\nSorry! Exiting program because of the error.\n");
                    printf("repeat  = %ld\n", repeat);
                    exit(1);
          }
 hfree(inCircularBuffer);
 hfree(history);
 hfree(weight);
 hfree(errStart);
```

```
 hfree(fX);
 hfree(cHat);
 hfree(error);
 hfree(gain);
}

/*************************************************************************/
/*FUNCTION get_data():
        This function reads values from the A/D process and places them into the
        array pointed to by the data pointer
        This function also sends the output data to the D/A process using the
        array pointed to by the output pointer
*************************************************************************/

void get_data(){
        output = y;
/* writing new values to output board   */
        status = AO_Write(OutDeviceNumber, 0, *output);
                if (status) ErrClean("AO_Write", status);
        status = AO_Write(OutDeviceNumber, 1, 0);
                if (status) ErrClean("AO_Write", status);

/* updating dac, i.e. converting new values to voltage outputs*/
        status = AO_Update(OutDeviceNumber);
                if (status) ErrClean("AO_Update", status);

/* getting next sequential block of data */
        status = DAQ_Monitor(InDeviceNumber, -1, 0, inNumChans, data, &newestPtIndex,
                                                                &daqStopped);
        if (status) ErrClean("DAQ_Monitor***", status);

/*  checking data to ensure isn't old data */
        while(junk == newestPtIndex){
                status = DAQ_Monitor(InDeviceNumber, -1, 0, inNumChans, data, &newestPtIndex,
                                                                &daqStopped);
                if (status) ErrClean("DAQ_Monitor***", status);
        }
        junk = newestPtIndex;
        *error = *(data + 1);
}
/*************************************************************************/
/*FUNCTION fir():
        This function applies a FIR filter (weight vector W) to the array of reference
        data (data array)
/*************************************************************************/
void fir(){

 *outAccum = ((float)(*data)) * (*W);

 for (i = 1; i < length; i++){
        if (data == history) data = endAdd - 1;
        else data -= 2;
        W++;
        *outAccum += ((float)(*data)) * (*W);
```

```
 }
 *output = (int)*outAccum;
 data = dataRef;
 W = weight;
}

/*************************************************************************/
/*FUNCTION LMS():
        This function applies the LMS algorithm to the weight
        array using the given data and the error data
/*************************************************************************/
void LMS(){
 XRefLMS = filteredX;

 MSE += ((double)*error * (double)*error - MSE)/100;
 MSX += ((double)*data * (double)*data - MSX)/100;

 for (i = 0; i < length; i++){
        *W -=((*gain) * ((float)(*error)) * (*XRefLMS));
        if (XRefLMS == fX) XRefLMS =  endRefX;
        else XRefLMS --;
        W++;
 }
 W = weight;
}
/*************************************************************************/
/* FUNCTION fx():
                this function filters the reference signal and is the fx in the fxLMS
                algorithm.  An FIR model of the secondary path is used and the weight
                values are contained in the C array.  cHat contains the premanent
                address to the start of the C array
/*************************************************************************/
void fx(){
 CdataRef = data;
 *filteredX = ((float)(*data)) * (*C);

 for (i = 1; i < Clength; i++){
        C++;
        if (data == history) data = endAdd - 1;
        else data -= 2;
        *filteredX += ((float)(*data)) * (*C);
 }
 C = cHat;
 data = CdataRef;
 //fprintf(outfile,"fX %d,   %p\n",*filteredX, filteredX);
}
/*************************************************************************/

/*  END OF ANR PROGRAM FUNCTIONS  */

/*************************************************************************/
```

APPENDIX D

FILTER CIRCUIT DIAGRAM



EMBRY-RIDDLE AERONAUTICAL UNIVERSITY		
ENGINEERING SUPERVISOR: JEROEN DOLMANS		
Title: ACTIVE FILTER		
Size: A	Document Number: ACTFIL10.SCH	REV: 1.0
Date: August 15, 1997	Sheet 1 of 1	

APPENDIX E

1. "AT-AO-6/10 Users Manual," National Instruments, May 1995.

2. "AT-MIO E Series Register-Level Programmer Manual," National Instruments, May 1995.

3. "AT-MIO E Series Users Manual," National Instruments, May 1995.

4. Bjarnason, E., "Algorithms for Active Noise Cancellation Without Exact Knowledge of the Error-Path Filter," Proceedings of IEEE International Symposium on Circuits and Systems, 1994.

5. Burdisso, R. A. and Thomas, R. H., "Active Control of Fan Noise From a Turbofan Engine," 31st Aerospace Sciences Meeting and Exhibit, January 1993.

6. Burrin, R. H., Khan, M. M. S., Salikuddin, M., and Tanna, H. K., "Application of Active Noise Control to Model Propeller Noise," *Journal of Sound and Vibration*, 1990.

7. "DAQ-STC™ Technical Reference Manual," National Instruments, May 1995.

8. Elliot, S. J., Nelson, P. A., Pinnington, R. J., and Thomas, E. R., "Active Control of Sound Transmission Through Stiff Lightweight Composite Fuselage Constructions," DGLR, AIAA Aeroacoustics Conference, May 1992.

9. Elliott, S. J., Stothers, I. M., and Nelson, P. A., "A Multiple Error LMS Algorithm and Its Application to the Active Control of Sound and Vibration," *IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-35, No. 10*, October 1987.

10. Elliot, S. J. and Tsujino, M., "A Globally Optimal Formulation for Feedforward Active Sound Control," Mechanical Systems and Signal Processing, 1991.

11. Hansen, C. H. and Snyder, S. D., "Convergence Characteristics of the Multiple Input, Multiple Output LMS Algorithm," *Journal of Intelligent Material Systems and Structures*, Vol. 3, January 1992.

12. Hansen, C. H. and Snyder, S. D., "The Influence of Transducer Transfer Functions and Acoustic Time Delays on the Implementation of the LMS Algorithm in Active Noise Control Systems," *Journal of Sound and Vibration*, 1990.

13. Nagel, R. T. and Sutliff, D. L., "Active Control of Far-Field Noise From a Ducted Propeller," 15th AIAA Aeroacoustics Conference, October 1993.

14. "NI-DAQ® Function Reference Manual for PC Compatibles," National Instruments, May 1995.

15. "NI-DAQ® User Manual for PC Compatibles," National Instruments, May 1995.

16. Sutliff, D. L., "Active noise control of the farfield noise radiated by a ducted fan," Ph.D. thesis, North Carolina State University, 1992.